

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

ZAVRŠNI RAD

Marko Pranjić

Zagreb, 2017.

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

ZAVRŠNI RAD

Mentor :

Prof. dr. sc. Mladen Crneković, dipl. ing.

Student :

Marko Pranjić

Zagreb, 2017.



SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE



Središnje povjerenstvo za završne i diplomske ispite
Povjerenstvo za završne ispite studija strojarstva za smjerove:
proizvodno inženjerstvo, računalno inženjerstvo, industrijsko inženjerstvo i menadžment, inženjerstvo
materijala i mehatronika i robotika

Sveučilište u Zagrebu Fakultet strojarstva i brodogradnje	
Datum	Prilog
Klasa:	
Ur.broj:	

ZAVRŠNI ZADATAK

Student: MARKO PRANJIĆ

Mat. br.: 0035195807

Naslov rada na hrvatskom jeziku: **MOBILNI ROBOT ZA SLIJEĐENJE LINIJE**

Naslov rada na engleskom jeziku: **MOBILE ROBOT FOR LINE TRACKING**

Opis zadatka:

Mobilni roboti zbog svoje fleksibilnosti imaju sve veću ulogu u modernoj proizvodnji materijalnih dobara. Služe za dostavu sirovina, poluproizvoda za ugradnju, alata te za odvoz gotovih proizvoda. Najpoznatiji princip njihovog rada i dalje je AGV (Automated Guided Vehicle) koji za navigaciju vozila koristi slijeđenje linije u obliku žice ili obojene trake.

Potrebno je projektirati model AGV-a s mogućnošću slijeđenja linije zadane trakom.

Tražena rješenja:

- konstruirati vozilo na principu diferencijalnog pogona
- odabrati motore, senzore i upravljački sustav
- ostvariti komunikaciju s nadređenim računalom
- definirati algoritam i eksperimentalno pokazati slijeđenje zadane trake

Zadatak zadan:

30. studenog 2016.

Zadatak zadao:

Prof.dr.sc. Mladen Crneković

Rok predaje rada:

- 1. rok: 24. veljače 2017.
- 2. rok (izvanredni): 28. lipnja 2017.
- 3. rok: 22. rujna 2017.

Predviđeni datumi obrane:

- 1. rok: 27.2. - 03.03. 2017.
- 2. rok (izvanredni): 30. 06. 2017.
- 3. rok: 25.9. - 29. 09. 2017.

v.d. predsjednik Povjerenstva:

Izv. prof. dr. sc. Branko Bauer

Izjavljujem da sam ovaj rad izradio samostalno koristeći znanja stečena tijekom studija i navedenu literaturu.

Marko Pranjić

SADRŽAJ

SADRŽAJ	I
POPIS SLIKA	II
POPIS TABLICA.....	III
POPIS OZNAKA	IV
SAŽETAK.....	V
SUMMARY	VI
1. UVOD	1
1.1. Primjeri modela mobilnog robota za slijeđenje linije.....	1
2. PRINCIP UPRAVLJANJA MOBILNOG ROBOTA.....	3
3. OPIS KORIŠTENIH KOMPONENTI.....	5
3.1. Stabilizator napona L7805.....	5
3.2. Integrirani krug L293d.....	7
3.3. Istosmjerni motor s reduktorom	9
3.4. Infracrveni senzori	10
3.5. Mikrokontroler – ATmega32.....	11
4. UPRAVLJAČKI SUSTAV MOBILNOG ROBOTA	14
5. MOGUĆA POBOLJŠANJA FUNKCIONALNOG I DIZAJNERSKOG SEGMENTA MOBILNOG ROBOTA	17
6. OPIS PROGRAMA (<i>Firmware</i>).....	18
7. ANALIZA REGULACIJE POZICIJE MOBILNOG ROBOTA	26
7.1. P regulator.....	27
7.1.1. Regulacija pozicije pri maksimalnoj brzini vozila.....	28
7.1.2. Regulacija pozicije vozila (40 % manja brzina).....	29
7.2. PD regulator.....	30
7.3. PID regulator	32
8. ZAKLJUČAK	33
LITERATURA.....	34
PRILOZI.....	35

POPIS SLIKA

Slika 1.1.	Pololu 3pi Robot i ActivityBot Robot	2
	Preuzeto sa: https://www.pololu.com/product/975 , pristup 13.9.2017.	
Slika 2.1.	Principijelna shema upravljanja mobilnog robota	3
Slika 2.2.	Diskretizacija pozicije mobilnog robota.....	4
Slika 3.1.	Osnovni spoj L7805 stabilizatora napona	5
Slika 3.2.	Utjecaj promjene struje tereta na regulaciju izlaznog napona L7805 stabilizatora	6
Slika 3.3.	L293d – raspored pinova	7
Slika 3.4.	Istosmjerni motor s reduktorom	9
Slika 3.5.	Električna shema modula IR senzora	10
	Preuzeto sa: https://goo.gl/images/jKGNfY , pristup 12.6.2017.	
Slika 3.6.	Mikroprocesor ATmega32 mikrokontrolera	12
Slika 4.1.	Električna shema mikroprocesorskog sustava.....	14
Slika 4.2.	Upravljačka jedinica mobilnog robota	16
Slika 5.1.	Mobilni robot za slijeđenje linije.....	17
Slika 7.1.	Elementi osnovnog regulacijskog kruga	26
Slika 7.2.	Staza na kojoj se provodi mjerenje.....	27
Slika 7.3.	Pozicija vozila sa P regulatorom (maksimalna brzina vozila).....	28
Slika 7.4.	Izlazna vrijednost P regulatora (maksimalna brzina vozila)	28
Slika 7.5.	Pozicija vozila sa P regulatorom (40 % manja brzina vozila).....	29
Slika 7.6.	Izlazna vrijednost P regulatora (40 % manja brzina vozila).....	29
Slika 7.7.	Pozicija vozila sa PD regulatorom	31
Slika 7.8.	Izlazna vrijednost PD regulatora	31

POPIS TABLICA

Tablica 2.1.	Devet mogućih vrijednosti pozicija robota.....	4
Tablica 3.1.	Električne i toplinske karakteristike L7805 regulatora.....	6
Tablica 3.2.	L293d – logika upravljanja.....	8
Tablica 3.3.	Električne i toplinske karakteristike – L293d.....	8
Tablica 3.4.	Tehničke karakteristike DC motora.....	9
Tablica 3.5.	Tehničke karakteristike infracrvenog senzora	11
Tablica 3.6.	Tehničke karakteristike ATmega32 mikrokontrolera.....	13

POPIS OZNAKA

Oznaka	Jedinica	Opis
e	-	Greška (razlika referentne i mjerene veličine)
f	Hz	Frekvencija mikroprocesora
i_m	-	Prijenosni omjer istosmjernog motora
I_o	A	Izlazna struja L7805 stabilizatora napona
I_s	mA	Struja napajanja infracrvenog senzora
$I_{o\text{ kont}}$	mA	Kontinuirana izlazna struja L293d <i>driver</i> -a
$I_{o\text{ pk}}$	A	Vršna izlazna struja L293d <i>driver</i> -a
I_m	mA	Struja napajanja istosmjernog motora
K_p	-	Proporcionalno pojačanje
K_i	-	Integralno pojačanje
K_d	-	Derivacijsko pojačanje
L	dB	Bučnost istosmjernog motora
M_m	Nmm	Moment istosmjernog motora
m	g	Masa motora
n_m	1/min	brzina vrtnje istosmjernog motora
P_D	W	Rasipana snaga L7805 stabilizatora napona
t	s	Vrijeme
U_m	V	Napon napajanja istosmjernog motora
U_s	V	Napon napajanja infracrvenog senzora
V_{CCI}	V	Napon logičkog kruga L293d <i>driver</i> -a
V_I	V	Ulazni napon L7805 stabilizatora napona
V_O	V	Izlazni napon L7805 stabilizatora napona
$y(t)$	-	Izlazna vrijednost regulatora u ovisnosti o vremenu
ϑ	°C	Područje radne temperature L7805 stabilizatora napona
ϑ_{JA}	°C/W	Toplinska otpornost L7805 elementa prema okolišu

SAŽETAK

Mobilni robot za slijeđenje linije je jedan od vrsti automatiziranih vođenih vozila (AGV) koji će biti razmatran u ovom seminaru. Automatski vođena vozila danas imaju veliku ulogu u logističkoj podršci mnogih firmi sa velikim skladištima i složenim sustavom proizvodnje. Takvi mobilni roboti omogućavaju veću produktivnost, odnosno postiže se veća efikasnost, što rezultira većim slobodnim kapacitetom u skladištu, te očekivanim brojem primljenih i poslanih paketa tokom jednog dana. U ovome području roboti su zamijenili ljudski rad iz više razloga, a neki od njih su: monoton posao, fizički napor te preciznost. Postoji više metoda kako mobilni roboti ostvaruju slijeđenje zadane putanje, a kao što je rečeno ovdje će se govoriti o slijeđenju obojene linije. U ovome radu naglasak će biti na ispitivanju različitih linearnih regulatora koji će se koristiti za upravljanje mobilnog robota. Napravljen je jednostavan model mobilnog robota za slijeđenje linije s mikrokontrolerom kao upravljačkom jedinicom, na kojem će se eksperimentalno potvrditi djelovanje različitih linearnih regulatora.

Ključne riječi: Mobilni robot, automatiziran, skladište, produktivnost, efikasnost, slijeđenje linije, linearni regulator, eksperimentalna potvrda.

SUMMARY

Mobile robot for line tracking is one type of the automated guided vehicles (AGV) which will be considered in this seminar. Today, automated guided vehicles have a big role in logistic support of many firms which have big warehouses and complicated manufacturing system. Such mobile robots are enabling better productivity, and higher efficiency, what results in a bigger free capacity of the warehouse, and in the expected number of received and shipped packages during one day. In this area robots have replaced human work due to many reasons, some of them are: monotonous work, physical effort and precision. There exists a lot of methods by which mobile robots can achieve to follow the predetermined path, and like it was said, here will be the talk about following coloured line. In this final thesis an emphasis will be on examination of diverse linear regulators which will be used for controlling the mobile robot. It was made a simple model of a mobile robot for line tracking with a microcontroller as a control unit, on which will be experimentally confirmed behaviour of a diverse linear regulators.

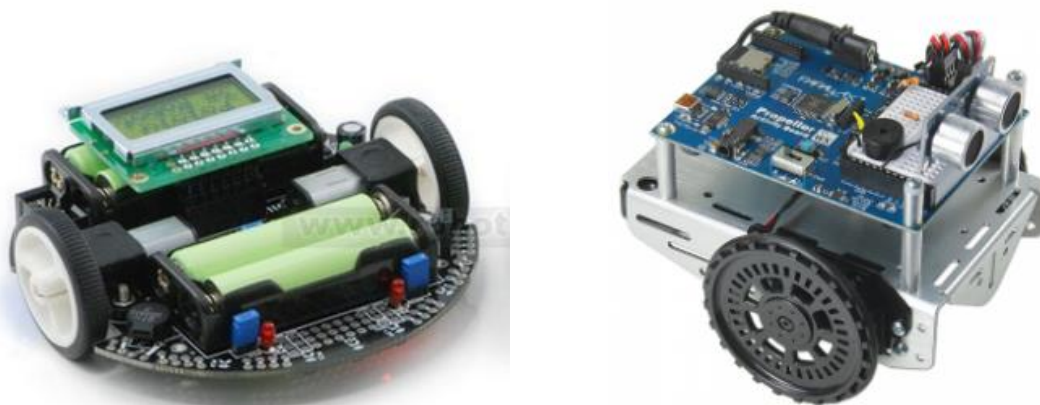
Key words: Mobile robot, automated, warehouse, productivity, efficiency, line tracking, linear regulator, experimental confirmation.

1. UVOD

Tvrtke sa serijskom proizvodnjom, odnosno velikim brojem proizvedene robe, imaju visoke zahtjeve na skladištenje robe, kao i zahtjeve na količinu isporučenih proizvoda tokom jednog dana. Kako bi se zadovoljili visoki zahtjevi kupaca i riješili logistički problemi, skladištima mnogih tvrtki upravljaju automatski vođena vozila, poznata pod skraćenicom AGV (Automated guided vehicle). Automatski vođeno vozilo je mobilni robot koji se koristi u svrhu transporta sirovina (npr. metal, plastika, guma), poluproizvoda koji moraju ići na naknadnu obradu i gotovih proizvoda. Neke od industrija koje koriste ovakva vozila su: prehrambena, farmaceutska, kemijska, automobilska. Mobilni roboti mogu se razlikovati prema metodi po kojoj se kreću u prostoru, vrsti pogona (diferencijalni pogon je uobičajen), te prema konstrukciji, odnosno vrsti proizvoda koje nose. Današnja automatski vođena vozila su najčešće laserski navođena. Reflektivne trake se postavljaju na zidove i nepomične strojeve, a na mobilnom robotu se nalazi rotirajuća kupola koja je ujedno laserski odašiljač i prijamnik. Mjeri se kut i udaljenost od nepomičnih objekata, te se izmjerena pozicija uspoređuje sa prostornom mapom koja je spremljena u memoriju upravljačkog sustava vozila. Također se koristi metoda slijeđenja linije u obliku žice koja se postavlja u pod, 20 – 30 mm ispod površine, gdje se utor reže duž putanje po kojoj se robot treba kretati. Slijeđenje putanje ostvaruje se na način da žica odašilje radio signal, a vozilo sa prikladnim senzorom koji je smješten blizu poda mjeri relativnu poziciju odašiljanog radio signala. Nedostatak ove metode je nefleksibilnost u smislu postavljanja nove putanje kretanja robota. U ovome radu će se opisivati najjednostavnija metoda, a to je slijeđenje linije u obliku obojane trake. Ako koristimo senzor sa infracrvenim zrakama, intenzitet reflektiranih zraka neće biti jednak prilikom odbijanja od svijetlu i tamnu podlogu. Upravo taj princip se koristi pri određivanju pozicije robota. Ovaj pristup rješava prethodni problem nefleksibilnosti jer se nove linije lako mogu označiti na podu. Problem ove metode je što se s vremenom linije mogu oštetiti i izlizati zbog velikog prometa i prljavštine. Naglasak u ovome seminaru biti će na ispitivanju ponašanja mobilnog robota sa različitim linearnim regulatorima (P, PD, PID). U svrhu eksperimentalne potvrde, napravljen je jednostavan model mobilnog robota za slijeđenje obojane linije. Infracrveni senzori će slati informaciju o poziciji robota AT Mega 32 mikrokontroleru kao upravljačkoj jedinici robota, a mikrokontroler zatim preko bluetooth veze šalje tu istu informaciju na računalo, na kojem će se dati grafički prikaz regulacije pozicije robota. U nastavku će se dati opis upravljačkog sustava robota, njegovih komponenti, kao i opis svih komponenti mobilnog robota. *Firmware* pisan u programskom jeziku C, koji upravlja radom robota, također će se opisati u daljnjem izlaganju.

1.1. Primjeri modela mobilnog robota za slijeđenje linije

Jedan od najuspješnijih mobilnih robota u slijeđenju linije je Pololu 3pi robot tvrtke Pololu Robotics & Electronics, čija konstrukcija omogućava brzo i precizno vođenje linijom. Pololu 3pi robot ima dva pogonska kotača i pcb ploču kružnog oblika koja ujedno čini i tijelo robota. Zbog kompaktne izvedbe robot se može okretati u mjestu, što ga čini prikladnim za korištenje u skučenim prostorima. Ovaj robot karakterizira i velika brzina, a maksimalna iznosi 1 m/s.



Slika 1.1. Pololu 3pi Robot i ActivityBot Robot

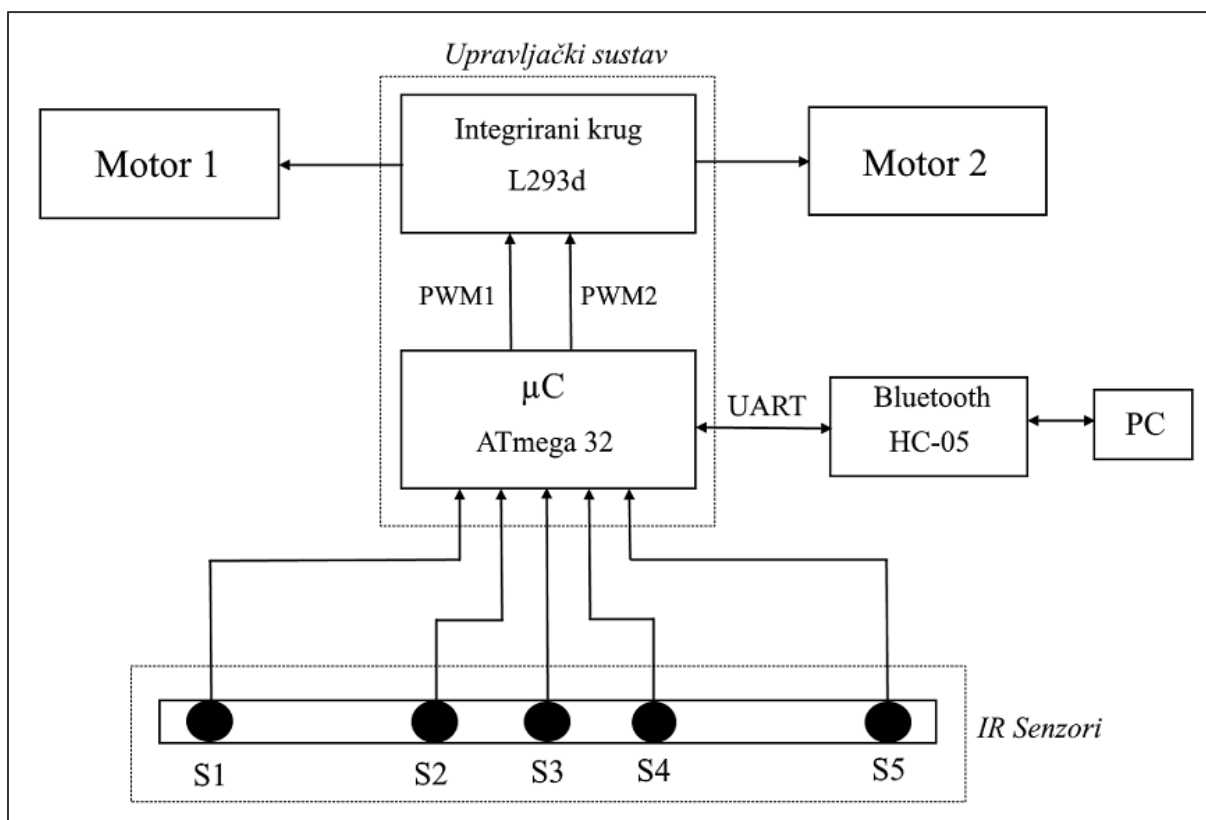
Kvalitetno vođenje robota po liniji osigurava pet optičkih senzora koji su raspoređeni u obliku luka. Za napajanje Pololu robota koriste se četiri NiMh AA baterije. Mobilni robot je upravlján ATmega 328P mikrokontrolerom koji se može programirati pomoću AVR USB programatora, te je po pitanju funkcionalne fleksibilnosti naklonjen korisniku. Nedostatak ovoga robota je mogućnost korištenja samo dva ulazno/izlazna digitalna pina koja preostaju korisniku, te nemogućnost komunikacije robota sa računalom.

ActivityBot mobilni robot tvrtke Parallax je izrađen za edukacijske svrhe, a posebno je pogodan za početnike u programiranju. Ovaj robot se može programirati u grafičkom sučelju ili pomoću C jezika. ActivityBot mobilni robot ima jednostavnu aluminijsku konstrukciju i dva pogonska kotača. Istosmjerni motori ovoga robota imaju enkodere koji osiguravaju brzo i precizno manevriranje robota. Ovo je modularan robot na kojeg se može nadograđivati razne elektroničke komponente, što znači da može obavljati i više funkcija poput: slijeđenje linije, izbjegavanje prepreka, praćenje svijetla itd. ActivityBot odlikuje jednostavno napajanje sa 5xAA baterija od 1.5 V.

Iako mnogi edukacijski roboti na tržištu nude mnogo fleksibilnosti po pitanju funkcionalnosti robota, odlučio sam napraviti vlastiti model mobilnog robota. Izradom električne sheme upravljačkog sustava, htio sam se približiti energetsom dijelu robota. Naredbe koje koristimo pri programiranju mikrokontrolera ovise o specifikacijama *hardware*-a kojim upravljamo, stoga za kvalitetno razumijevanje je bitno dovesti u korelaciju ove dvije komponente. Također, sve funkcije pinova mikrokontrolera su na raspolaganju, dok gore navedeni roboti imaju određeni broj slobodnih pinova koje nude korisniku. Praktičnim radom dobivaju se i nova znanja kao što su: sposobnost analize greški, što je uzrokovalo njihovu pojavu, te otklanjanje istih. Komunikacija upravljačkog sustava robota sa računalom je bitna zbog analize ispravnosti rada robota, a to ovaj robot i posjeduje. Nedostatak vlastitog modela mobilnog robota u odnosu na gore navedene je neučinkovito napajanje, gdje rad robota ovisi o padu napona na baterijama. Kompaktna konstrukcija i dizajn, također su prednosti gore navedenih robota.

2. PRINCIP UPRAVLJANJA MOBILNOG ROBOTA

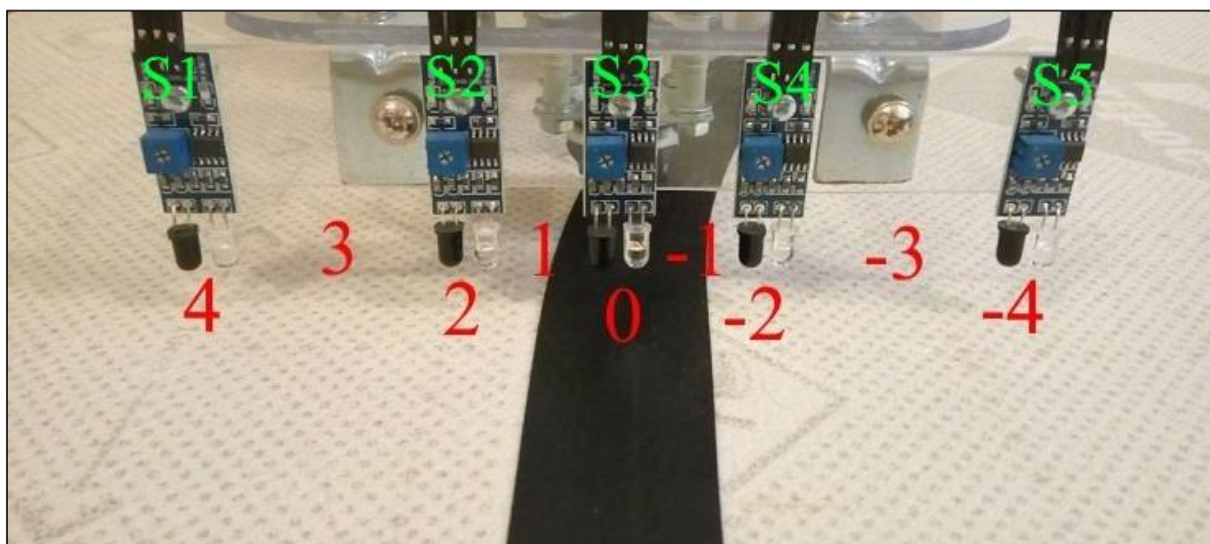
Princip upravljanja mobilnog robota je prikazan simboličkom shemom na slici 2.1.



Slika 2.1. Principijelna shema upravljanja mobilnog robota

Mobilni robot za slijeđenje linije se sastoji od tri osnovna dijela: senzora, upravljačkog dijela i istosmjernih motora. Pet infracrvenih senzora S1...S5 su postavljeni 1 cm od poda, raspoređeni kao što je prikazano slikom 2.1. Na sredini letvice je postavljen manji razmak između senzora, jer je na tome mjestu potrebna veća rezolucija pri mjerenju pozicije kako bi zadržali mobilnog robota na zadanoj liniji. Više informacija o infracrvenim sensorima i njihovim izlazima, nalazi se u potpoglavlju 3.4. Informacija o poziciji robota se šalje sa senzora na mikrokontroler, gdje je diskretizacija pozicije po sensorima prikazana na slici 2.2. Mikrokontroler registiranu poziciju uspoređuje sa referentnom pozicijom koja je zapisana u programu. Na temelju greške, regulator koji je implementiran u programu će dati određenu izlaznu vrijednost. Na temelju programa koji je detaljnije opisan u poglavlju 5., mikrokontroler će dati dva PWM signala koje šalje na L293d H-most. H-most daje napon određenog iznosa na motor „Motor 1“ i motor „Motor 2“, koji su proporcionalni faktoru popunjenosti PWM1, odnosno PWM2 signala. Variranjem faktora popunjenosti dva PWM signala, omogućen je diferencijalni pogon.

Upravljački sustav mobilnog robota ima mogućnost komunikacije sa računalom, ili drugim uređajima koji podržavaju bluetooth komunikaciju. Informacija o poziciji robota se prvotno šalje sa mikrokontrolera putem serijske UART veze do HC-05 bluetooth modula, a zatim HC-05 modul, bluetooth vezom šalje informaciju na računalu. Ova komunikacija je također dvosmjerna, što znači da osobnim računalom možemo upravljati radom robota. Nadzor podataka pomoću osobnog računala je bitan zbog utvrđivanja kvalitete regulacije, i općenito zbog utvrđivanja ispravnosti sustava kojim upravljamo.



Slika 2.2. Diskretizacija pozicije mobilnog robota

Tablicom 2.1. je prikazano devet mogućih pozicija robota koje su definirane programom (vidi 5. poglavlje). Aktivni senzori su senzori koji „vide“ crnu liniju.

Tablica 2.1. Devet mogućih vrijednosti pozicija robota

Aktivni senzori	Pozicija
S1	4
S1 i S2	3
S2	2
S2 i S3	1
S3	0
S3 i S4	-1
S4	-2
S4 i S5	-3
S5	-4

3. OPIS KORIŠTENIH KOMPONENTI

3.1. Stabilizator napona L7805

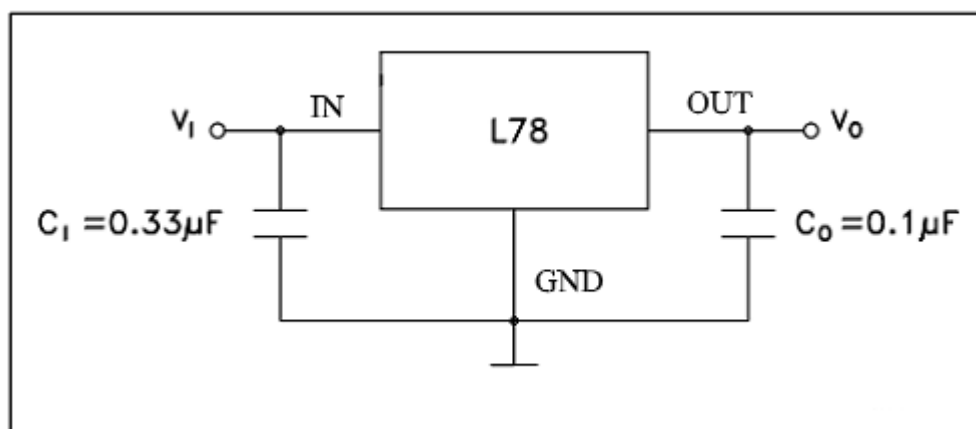
Integrirani krug L7805 je stabilizator napona kojemu je zadatak osigurati stalan i stabilan napon na svim uređajima koje napaja. Izrađuje se u više izvedbi, a u ovom projektu je korišteno TO-220 pakovanje, proizvođača ST Microelectronics. Postoji više serija L78 stabilizatora napona, gdje serija 7805 daje izlazni napon od 5 V kako sugeriraju dvije zadnje brojke u imenu. Osim što su ovi integrirani krugovi namijenjeni da osiguraju fiksirani stabilan napon, uz modifikaciju i dodavanje vanjskih komponenti može se dobiti prilagodljiv izlazni napon [3]. L7805 ima i ugrađenu zaštitu od toplinskog preopterećenja, te limitiranje maksimalne struje kroz element. Kako bi odredili trebamo li koristiti hladnjak za navedeni element, moramo izračunati disipaciju snage. Disipirana energija pretvara se u toplinu koja zagrijava element, gdje točnost regulacije L7805 integriranog kruga opada sa porastom temperature. Rasipanu snagu možemo izračunati tako da pomnožimo razliku ulaznog i izlaznog napona sa izlaznom strujom elementa L7805, prema sljedećoj formuli :

$$P_D = (V_I - V_O) * I_O, \quad (3.1)$$

gdje je:

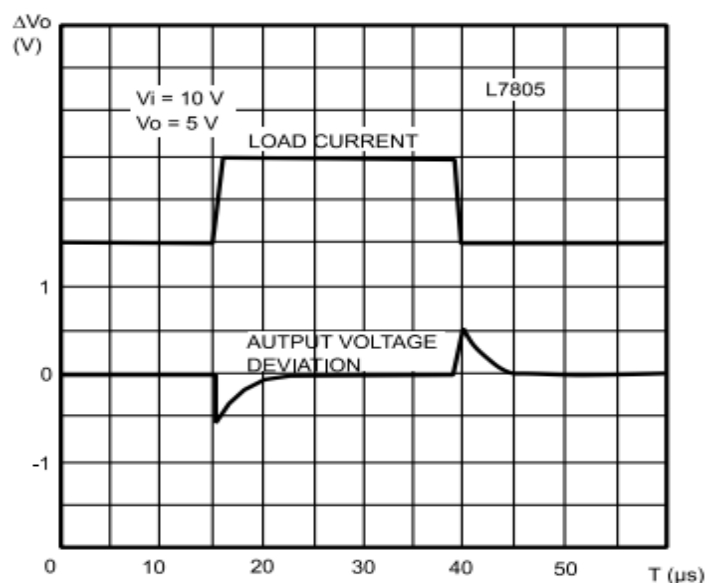
- P_D - iznos rasipane snage [W]
- V_I - ulazni napon L7805 regulatora (min. 7 V)
- V_O - izlazni napon L7805 regulatora (približno 5 V)
- I_O - izlazna struja L7805 regulatora

Za izračun rasipane snage elementa L7805 prema shemi sa slike 2.1. , uvršteni su konkretni podatci : $V_I = 10.5$ V, $V_O = 5$ V i $I_O = 0.33$ A. Tako se dobilo da je $P_D = 1.55$ W, što prema [3] znači da ne moramo koristiti hladnjak. Unatoč tome hladnjak se može staviti kako bi bili na strani sigurnosti i omogućili dug vijek trajanja elementa.



Slika 3.1. Osnovni spoj L7805 stabilizatora napona [3]

Kao što se vidi na slici 3.1., L7805 ima tri izvoda (input, GND, output). Na ulazni pin dovodi se napon koji se želi stabilizirati, na izlaznom pinu se dobiva stabilizirani napon +5V, a na izvod GND se spaja hladnjak koji preuzima toplinu sa elementa. Između ulaznog pina V_I i GND spaja se keramički ili tantal kondenzator ($0.33 \mu\text{F}$) kojim stabiliziramo ulazni napon. Ovaj kondenzator se koristi zbog niske impedancije i pogodnosti rada sa visokim frekvencijama koje su prisutne pri napajanju komponenti kao što je mikrokontroler. Prilikom velikih promjena struje tereta, izlazni napon L7805 regulatora će oscilirati, gdje se prijelazna pojava smirivanja napona mjeri u mikrosekundama [μs]. Kako bi L7805 ostvario što bolju regulaciju izlaznog napona, odnosno imao bolji prijelazni odziv, na izlazni pin dodaje se još jedan keramički ili tantal kondenzator vrijednosti 100 nF .



Slika 3.2. Utjecaj promjene struje tereta na regulaciju izlaznog napona L7805 stabilizatora [3]

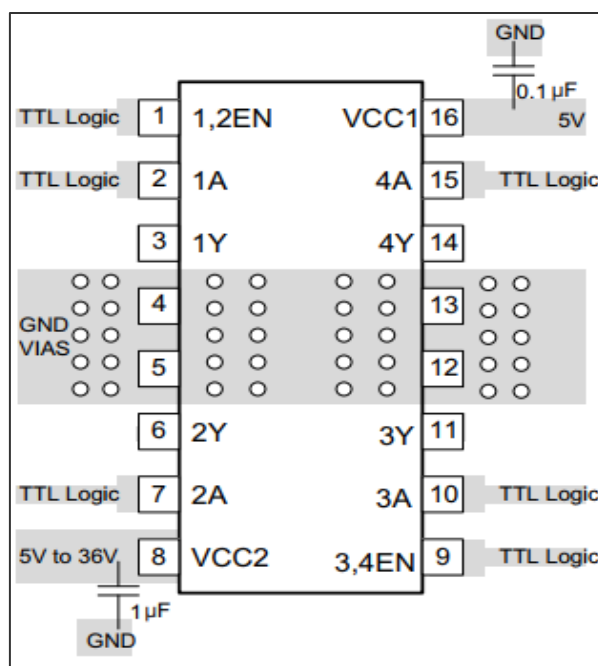
Električne i toplinske karakteristike L7805 regulatora dane su u tablici 3.1., detaljnije informacije mogu se pronaći u [3].

Tablica 3.1. Električne i toplinske karakteristike L7805 regulatora [3]

Simbol	Parametar	Min.	Typ.	Max.	Mjerna jedinica
V_I	Ulazni napon	7	-	25	V
V_O	Izlazni napon	4,75	5	5,25	V
I_O	Izlazna struja	5	-	1500	mA
ϑ	Radna temperatura	0	-	125	$^{\circ}\text{C}$
ϑ_{JA}	Toplinska otpornost prema okolišu	50			$^{\circ}\text{C}/\text{W}$

3.2. Integrirani krug L293d

L293d je četverostruki polu H-most ili dvostruki H-most u integriranoj izvedbi. H-most je aktuatorski sklop koji omogućava uređajima nižeg energetskeg nivoa signala da upravljaju trošilima većih snaga. Osim toga H-mostom se može postići promjena smjera toka struje kroz trošilo, što se najviše primjenjuje kod električnih motora kojima mijenjamo smjer vrtnje. L293d je namijenjen za pogon induktivnih trošila kao što su releji, solenoidi, DC i koračni motori, kao i za druga trošila visoke struje/napona. Komponenta se izrađuje u PDIP izvedbi, i ima 16 izvoda kako je prikazano na slici 3.3. Četiri *driver-a* (1A, 2A, 3A, 4A) upravljaju stanjima na pripadajućem izlazu (1Y, 2Y, 3Y, 4Y). *Driver-i* se aktiviraju u paru, gdje za aktiviranje para (1A, 2A) moramo na pin 1 (1,2 EN) dovesti 5 V, a za aktivaciju drugog para (3A, 4A) moramo pin 9 (3,4 EN) dovesti u visoko stanje. Prethodno navedeni pinovi su ulazni pinovi i pripadaju skupini TTL logike, zbog čega je energetske nivo signala mikrokontrolera kompatibilan sa L293d integriranim krugom. Ovaj element također sadrži povratne diode za zaštitu logičkih sklopova od reverzne struje koje znaju prouzročiti induktivna trošila.



Slika 3.3. L293d – raspored pinova [4]

Napajanje logičkog sklopa odvojeno je od napajanja trošila. Na pin 16 (VCC1) se uobičajeno dovodi 5 V za napajanje logičkog sklopa, dok se putem pina 8 (VCC2) napajaju trošila napona 5 V – 36 V. Na pinove 1 i 9 se obično dovodi PWM signal, a proporcionalno faktoru popunjenosti PWM signala, na izlazima (1Y, 2Y, 3Y, 4Y) će se pojaviti odgovarajući napon. Važno je napomenuti ako pinovi 1 i 9 nisu u visokom stanju, pripadajući izlazi će biti u stanju visoke impedancije bez obzira na stanje *driver-a* (1A, 2A, 3A, 4A).

U ovom projektu gledajući shemu na slici 2.1., PWM izlazi (OC1B, OC1A) AT Mega 32 mikrokontrolera su spojeni na pinove 1 i 9. *Driver-i* (1A, 2A, 3A, 4A) su također upravljani digitalnim izlazima mikrokontrolera, a logika upravljanja je dana u tablici 3.2. za slučaj ako je spojen električni motor na izlazu. Na izlaz (1Y, 2Y), odnosno (3Y, 4Y) su spojeni DC motori koji su opisani u potpoglavlju 3.3. Logički sklop je napajan preko pina 16 iz istoga izvora koji se spaja na konektor „J1“.

Tablica 3.2. L293d – logika upravljanja [4]

1,2 EN (3,4 EN)	1A (3A)	2A (4A)	Izlaz 1Y,2Y (3Y,4Y)
0	X ⁽¹⁾	X ⁽¹⁾	Stanje visoke impedancije
1	1	0	Jedan smjer vrtnje
1	0	1	Drugi smjer vrtnje
1	0	0	Stop
1	1	1	Stop

(1) Bilo koje stanje

Bitan parametar kod rada sa H-mostom je sklopna frekvencija, to je frekvencija PWM signala koji se dovodi na već spomenute pinove L293d čipa. Frekvenciju PWM-a definira mikrokontroler. Visoke frekvencije (>20 kHz) su pogodne za suzbijanje vibracija u električnim motorima pri nižim brzinama vrtnje, ali isto tako viša sklopna frekvencija izaziva prekomjerno zagrijavanje tranzistora L293d integriranog kruga. Zbog toga, takve elektroničke komponente imaju definiranu maksimalnu sklopnu frekvenciju od strane proizvođača, a za L293d to je 5 kHz. Izlazni napon je obično oko 1.4 V manji od VCC2 napona, pri čemu se rasipana energija pretvara u toplinu i također zagrijava element. Preporučeno je staviti hladnjak koji se spaja na GND pinove, ili omogućiti dovoljan sloj bakra na tiskanoj pločici gdje se spajaju GND pinovi.

Tablica 3.3. Električne i toplinske karakteristike - L293d [4]

Simbol	Parametar	Min.	Typ.	Max.	Mjerna jedinica
VCC1	Napon logičkog kruga	4.5	-	7	V
VCC2	Napon trošila	VCC1	-	36	V
I_{okont}	Kontinuirana izlazna struja	-600	-	600	mA
I_{opk}	Vršna izlazna struja	-1.2	-	1.2	A
V_o	Izlazni napon	VCC2 -1,8	VCC2 - 1,4	-	V
ϑ	Radna temperatura	0	-	70	°C
ϑ_{JA}	Toplinska otpornost prema okolišu	36,4			°C/W

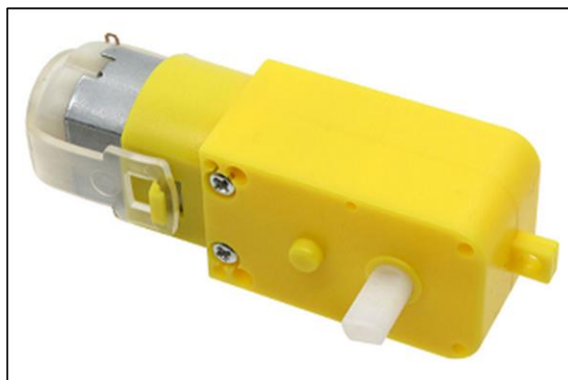
Toplinska otpornost prema okolišu ϑ_{JA} definira broj koji govori za koliko će temperatura u elementu biti veća od temperature okolišnog zraka, pri disipaciji snage od 1 W.

3.3. Istosmjerni motor s reduktorom

Za pogon mobilnog robota izabrana su dva istosmjerna motora male snage. Motori su izabrani zbog svoje jednostavnosti, velikog prijenosnog omjera, male potrošnje električne energije i zbog kompatibilnosti sa L293d integriranim krugom. Prema električnoj shemi na slici 2.1., jedan motor se spaja na konektor (X3-1, X3-2), a drugi motor na konektor (X4-1, X4-2). Navedeni konektori su redne stezaljke, a više o odabiru i rasporedu komponenti na tiskanoj pločici će biti riječi u potpoglavlju 3.6. Izvor električne energije za motore se spaja na konektor (X51, X52) u iznosu od 7 V (Treba uračunati pad napona od 1,4 V na izlazima L293d H-mosta). Ovi motori mogu raditi u području 3 V – 6 V, a iznos napona na stezaljkama motora ovisi o faktoru popunjenosti PWM signala koje daje mikrokontroler. L293d H-most „sjecka“ napon koji napaja istosmjerne motore u ritmu pulsno širinske modulacije (5 kHz), dobiva se srednja vrijednost napona „isjeckanog“ signala, te se na taj način može dobiti proizvoljna vrijednost napona napajanja motora. Električnim motorima „isjeckani“ napon je zapravo kontinuiran zbog njihove induktivne tromosti. Kako je to i karakteristično za istosmjerne motore, povećanjem napona armature, proporcionalno se povećava i njihova brzina vrtnje. Upravo zbog jednostavne promjene brzine vrtnje, istosmjerni motori su često korišteni. Oba motora su zasebno upravljana, čime se lako ostvaruje diferencijalni pogon. Motori su upravljani prema logici navedenoj u tablici 3.2. Tehničke karakteristike istosmjernih motora dane su u tablici 3.4. Izgled korištenog DC motora za pogon mobilnog robota dan je na slici 3.4.

Tablica 3.4. Tehničke karakteristike DC motora

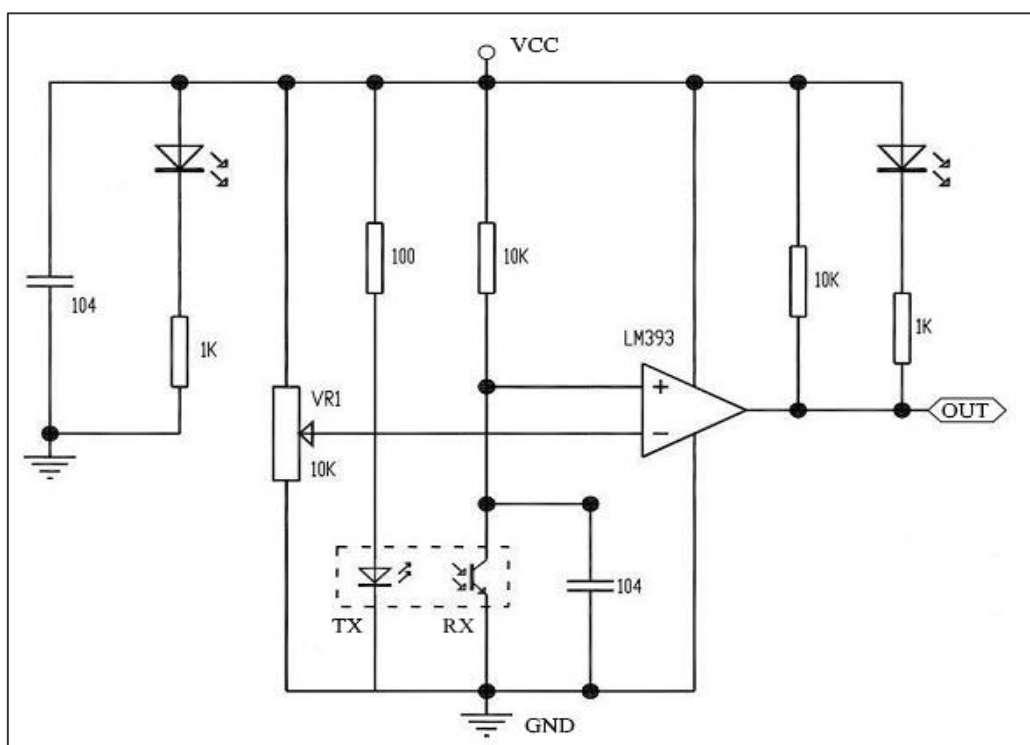
Parametar	Vrijednost	Mjerna jedinica
Napon, U_m	3 - 6	V
Struja, I_m	100 - 150	mA
Prijenosni omjer, i_m	48:1	-
Brzina vrtnje (bez tereta), n_m	100 - 240	1/min
Masa, m	50	g
Moment, M_m	78,5 – 107,9	Nmm
Bučnost, L	< 65	dB



Slika 3.4. Istosmjerni motor s reduktorom

3.4. Infracrveni senzori

Za osjetilni dio mobilnog robota izabrano je pet modula infracrvenih senzora. Senzori su odabrani zbog fleksibilnosti (moguće podešavanje osjetljivosti senzora), jednostavnosti, digitalnog izlaza sa električnim signalom iz TTL skupine, i cijene. Prema električnoj shemi na slici 2.1., pet senzora se spaja na 5x2 konektor „CON1“. Modul IR senzora ima tri izvoda (VCC, GND, OUT) od kojih su dva za napajanje senzora i jedan digitalni izlaz. Kako izvodi IR modula nisu kompatibilni sa konektorom „CON1“, napravljen je međukonektor koji omogućava vezu pet senzora sa upravljačkim sustavom preko flah kabela. Princip rada IR modula može se objasniti njegovom električnom shemom koja je prikazana na slici 3.5.



Slika 3.5. Električna shema modula IR senzora

Komparator LM393 omogućava pretvorbu analognih ulaza u digitalni izlaz. Izlaz komparatora će biti u visokom stanju samo kada je napon na neinvertirajućem ulazu veći od napona na invertirajućem ulazu. Napon na invertirajućem ulazu je referentni, a njegova vrijednost se može podešavati potencijetrom. IR modul se sastoji od led diode i fototranzistora (TX/RX par), gdje TX led dioda zrači infracrvenu svjetlost, a RX fototranzistor prima odbijene infracrvene zrake. Kada se senzor nalazi pod bijelom podlogom, intenzitet reflektiranih zraka je veći, te foto tranzistor provodi električnu energiju. To rezultira niskim naponom na neinvertirajućem ulazu komparatora ($\sim 0V$), odnosno izlaz je u niskom stanju ($1.8V$, $9\mu A$). U slučaju kada se senzor nalazi na crnoj liniji, intenzitet reflektiranih zraka je manji, a fototranzistor provodi malu tzv. tamnu struju (desetak nA). Napon neinvertirajućeg ulaza komparatora sada je jednak naponu između kolektora i emitera fototranzistora, koji je veći od referentnog napona, pa je i izlaz komparatora u visokom stanju ($\sim 4V$, $90\mu A$). Preko potencijetra „VR1“ na slici 3.5. možemo podesiti osjetljivost senzora, tako što mijenjamo referentni napon komparatora. Ako želimo da senzor može detektirati crnu liniju samo na maloj udaljenosti od nje, moramo podesiti

što veći referentni napon na invertirajućem ulazu LM393 komparatora. Digitalni izlazi senzora su spojeni na digitalne ulaze mikrokontrolera (PB0...PB4). Tehničke karakteristike senzora se nalaze u tablici 3.5. Izlazne struje niskog i visokog stanja senzora su izmjerene u spoju sa ulaznim pinovima ATmega32 mikrokontrolera.

Tablica 3.5. Tehničke karakteristike infracrvenog senzora

Parametar	Vrijednost	Mjerna jedinica
Napon napajanja, U_s	3,3 V - 5 V	V
Struja napajanja, I_s	40	mA
Izlazni napon niskog stanja	1,8	V
Izlazna struja niskog stanja	9	μ A
Izlazni napon visokog stanja	4	V
Izlazna struja visokog stanja	90	μ A
Područje detekcije	2 ~ 30	cm

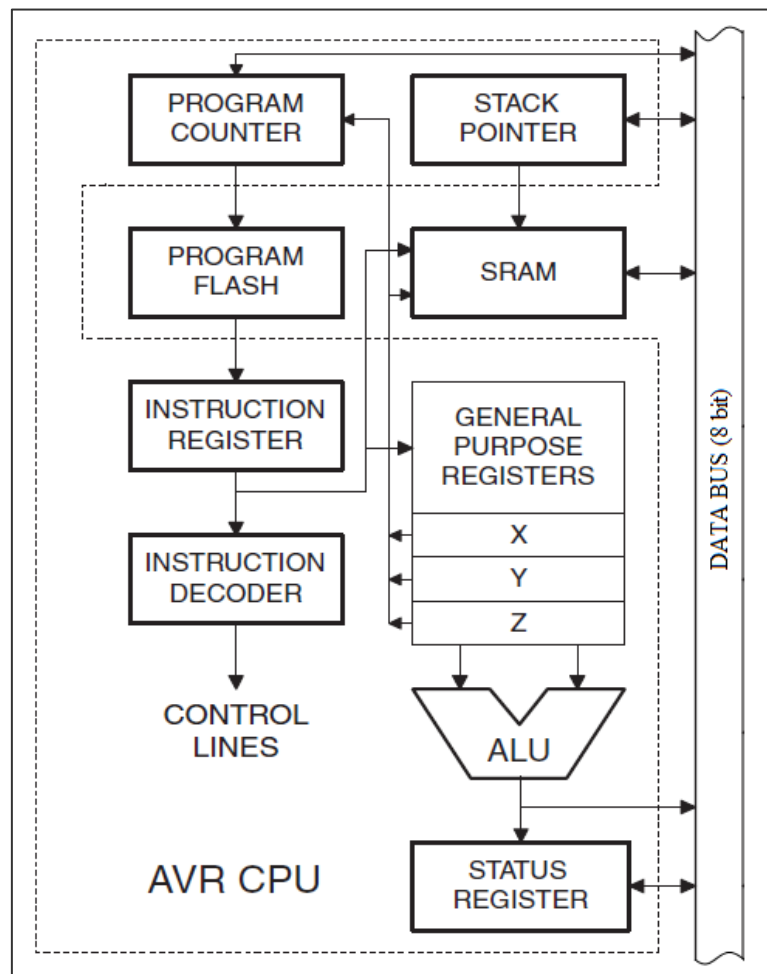
3.5. Mikrokontroler – ATmega32

Mikrokontroler ili mikroračunalo je digitalni automat sposoban za izvršenje računskih operacija na osnovu programa pohranjenog u memoriji [2]. Primjena mikrokontrolera je široka, a općenito su namijenjeni za digitalno upravljanje i regulaciju raznih procesa i strojeva. Česta primjena je u automobilske industriji, kućnim automatskim sustavima, industrijskoj automatizaciji, te u manjim mobilnim elektroničkim uređajima. Ovakva mikroračunala karakterizira mala potrošnja energije, fleksibilnost (zamjena *firmware-a*), dug vijek trajanja i niska cijena. Mikroprocesor zajedno sa perifernim jedinicama (ulazno - izlazna jedinica, ROM i RAM memorija, brojači, vremenski sklop - *timer*, analogno - digitalni pretvornici,...) čini mikrokontroler. Glavna funkcija mikroprocesora je osigurati pravilno izvođenje programa., te zbog toga mora biti u mogućnosti pristupiti memorijama, izvoditi računske operacije i upravljati perifernim jedinicama. Arhitektura mikroprocesora ATmega32 mikrokontrolera dana je na slici 3.6. Mikroprocesor se sastoji od upravljačke i aritmetičko logičke jedinice. Osnovne komponente μ P-a biti će opisane u nastavku:

- **Registar insrukcija** (*Instruction register*) dohvaća instrukcije spremljene u flash memoriji, dekodira ih u strojni jezik, te putem upravljačkih sabirnica upravlja aritmetičko logičkom ili perifernim jedinicama ovisno o vrsti instrukcije.
- **Programsko brojilo** (*Program counter*) je registar posebne namjene koji pokazuje do kuda se program izvršio, odnosno sadrži adresu na kojoj će se tražiti sljedeća instrukcija iz memorije.
- **Pokazivač složaja** (*Stack pointer*) je registar posebne namjene uglavnom korišten za spremanje privremenih podataka, lokalnih varijabli i za spremanje povratnih adresa pri pozivu potprograma [5]. Neposredno prije skoka na potprogram, na složaju za pozive (*call stack*) se stavlja adresa sljedeće instrukcije, dobavljena iz programskog brojila. Nakon izvršenja pozvanog potprograma, zadnja pohranjena adresa u složaju se briše (povratna adresa) i na nju se skače, te se na taj način program nastavlja izvršavati nakon

mjesta poziva potprograma. Registar je ograničen po pristupu svojim elementima zbog podatkovne strukture LIFO (*last in, first out*).

- **Registri opće namjene** (*General purpose registers*) služe za privremenu pohranu podataka koji se šalju u aritmetičko logičku jedinicu na obradu, te za spremanje rezultata te obrade. ATmega 32 ima 32 takva registra, svaki po 8 bit-ova.
- **Aritmetičko logička jedinica** (ALU) radi u direktnoj vezi sa svim 32 registra opće namjene. ALU operacije su podijeljene u tri glavne kategorije - aritmetičke (zbrajanje, oduzimanje, množenje, dijeljenje), logičke (I, ILI, NE...) i operacije na razini bita.
- **Statusni registar** (*Status register*) sadrži informaciju o rezultatu zadnje izvršene aritmetičke instrukcije, poslane od strane aritmetičko logičke jedinice.



Slika 3.6. Mikroprocesor ATmega32 mikrokontrolera [5]

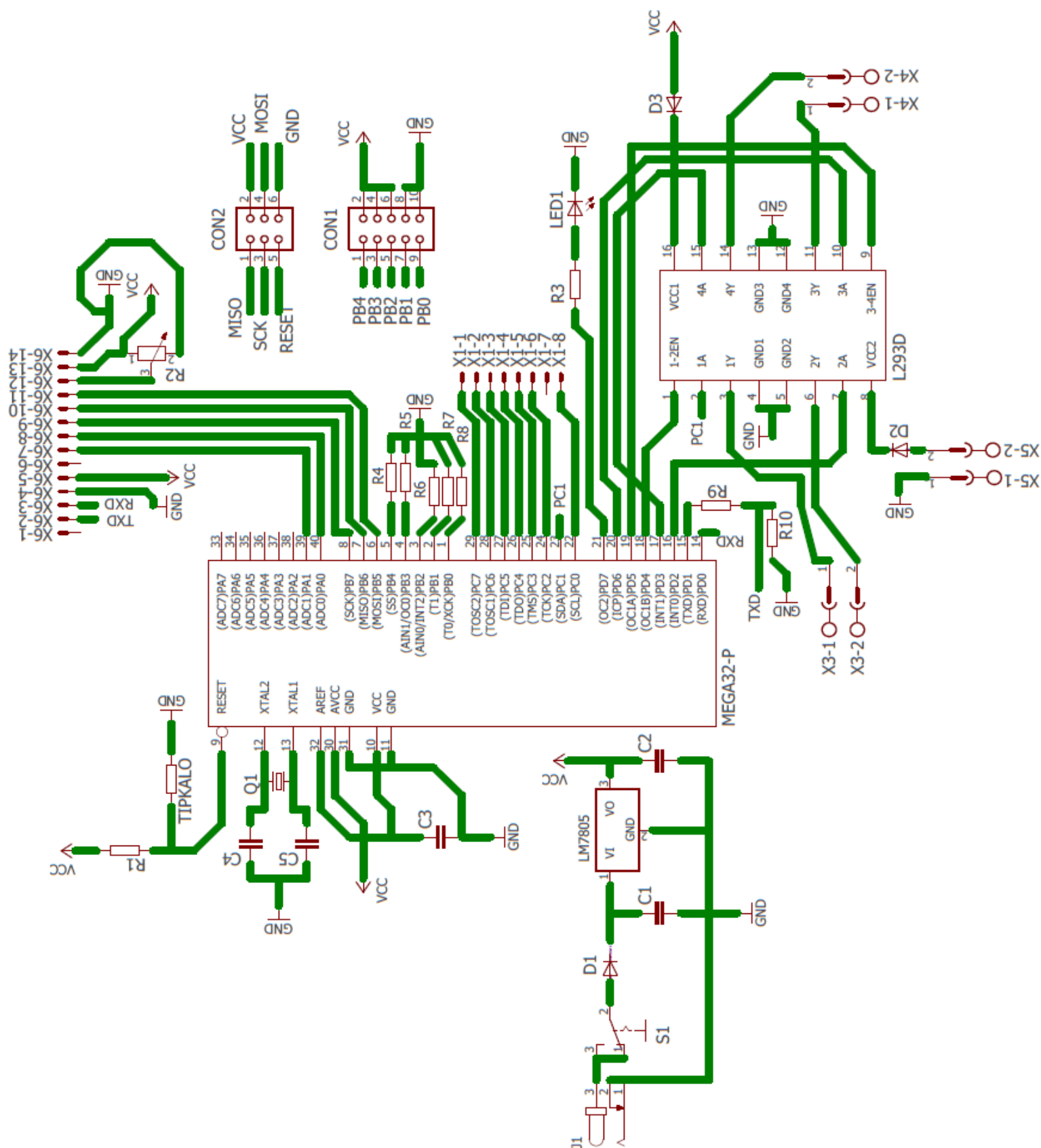
Podatkovna sabirnica služi za razmjenu podataka (u ovom slučaju 8 bit-ni), odnosno povezuje mikroprocesor sa perifernim jedinicama. Pomoću adresne sabirnice se adresiraju memorije, a upravljačka sabirnica kontrolira čitanje/pisanje (R/W) registara. Atmel-ov ATmega32 mikrokontroler (8 bit-ni) je niskopotrošni CMOS integrirani krug baziran na AVR poboljšanoj RISC arhitekturi. RISC (*Reduced instruction set computer*) arhitektura mikroprocesora ima jednostavniji skup instrukcija, gdje je za izvršenje jedne instrukcije potrebno 1-2 koraka takta. AVR-ov mikroprocesor karakterizira velika brzina izvršavanja instrukcija, te postiže 1 MIPS

(*Million instructions per second*) po 1 MHz-u [5]. Takt potreban za rad mikroprocesora može se u mikrokontroleru generirati na više načina, a na slici 2.1. se može vidjeti da je u ovom projektu spojen vanjski kristal 16 MHz koji pobuđuje oscilator u mikrokontroleru. ATmega32 sadrži periferne jedinice kao što su ulazno-izlazne jedinice (spajanje sa vanjskim procesima ili strojevima), RW (*read/write*) i RAM memorija, 8 analogno-digitalnih pretvornika (10 bit-ni), 4 PWM izlaza, SPI//TWI/UART serijska komunikacija, dva 8 bit-na i jedan 16 bit-ni vremenski sklop/brojač (*timer/counter*). Raspored i funkcije pinova ATmega32 mikrokontrolera se mogu vidjeti u [5] na strani 2. Tehničke karakteristike mikrokontrolera dane su u tablici 3.6.

Tablica 3.6. Tehničke karakteristike ATmega32 mikrokontrolera [5]

Parametar	Min.	Typ.	Max.	Mjerna jedinica
Napon napajanja	4,5	5	5,5	V
Struja napajanja	-	30	200	mA
Izlazni napon bilo kojeg pina (PORT A, B, C, D)	4,2	5	-	V
Izlazna struja bilo kojeg pina (PORT A, B, C, D)	-	20	40	mA
Frekvencija mikroprocesora, f	0	-	16	MHz
Flash memorija	32			kB
EEPROM	1024			B
SRAM	2			kB

4. UPRAVLJAČKI SUSTAV MOBILNOG ROBOTA



Slika 4.1. Električna shema mikroprocesorskog sustava

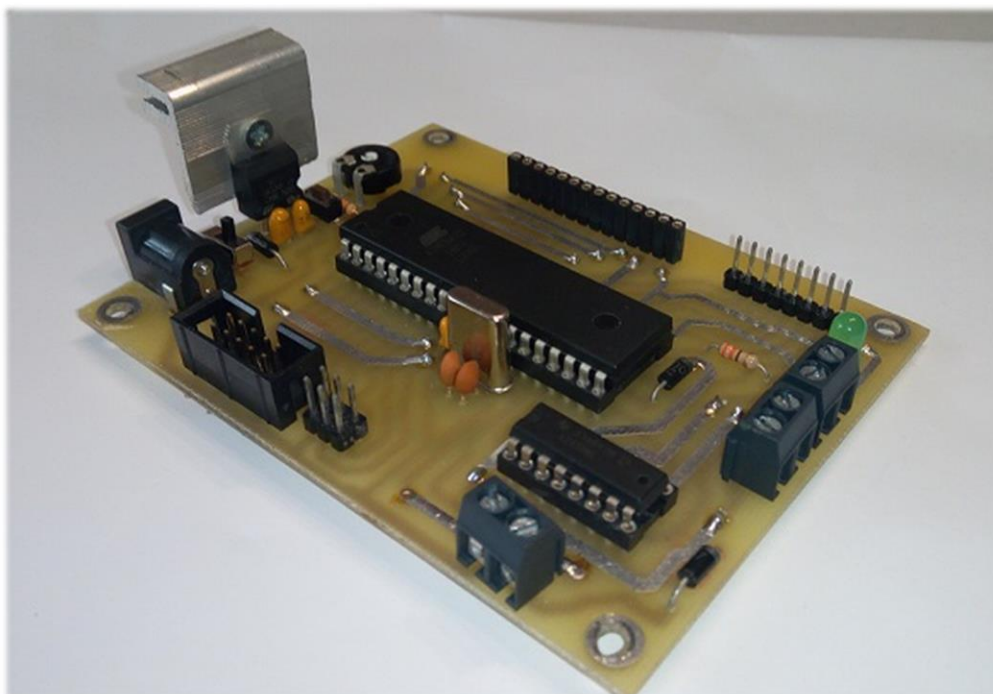
Mikroprocesorski sustav s ATmega32 mikrokontrolerom i L293d integriranim krugom čini upravljačku jedinicu mobilnog robota. Zadatak mikrokontrolera je prikupljati informacije sa senzora, te uz odgovarajući *firmware* obraditi te informacije i upravljati motorima. Opis ATmega32 mikrokontrolera i njegovih mogućnosti, detaljnije su opisani u potpoglavlju 3.5. Tiskana pločica je rađena u programu Eagle, a električna shema sa svim komponentama je prikazana na slici 4.1.

Kratak opis električne sheme dan je u nastavku, sve oznake se referiraju prema slici 4.1. :

- preko „J1“ DC konektora napaja se sva elektronika mobilnog robota (mikrokontroler, L293d, infracrveni senzori), a ulazni napon može biti u rasponu 8 V - 25 V, gdje je ukupna potrošnja struje jednaka 330 mA
- zbog moguće krive polarizacije ulaznog napona, dioda „D1“ se koristi kao zaštita, odnosno definira smjer toka struje
- integrirani krug L7805 se koristi za stabilizaciju izlaznog napona od 5 V, detaljniji opis dati će se u potpoglavlju 3.1.
- pinovi mikrokontrolera (PB0, PB1, PB2, PB3, PB4) se spajaju na izlaze infracrvenih senzora, te primaju vrijednosti sa senzora u digitalnom obliku „0“ ili „1“
- otpornici (R4, R5, R6, R7, R8) su „pull-down“ otpornici (47 k Ω) koji definiraju nisko stanje digitalnih ulaznih pinova (PB0, PB1, PB2, PB3, PB4), a samo u slučaju visokog stanja na jednom od izlaza senzora i pripadni ulazni pin će imati visoko stanje
- konektor „CON1“ služi kao veza između mikrokontrolera i pet senzora, detaljniji opis senzora dati će se u potpoglavlju 3.4.
 - izlazi senzora spajaju se na pinove 1, 3, 5, 7 i 9 konektora „CON1“, gdje se izlaz srednjeg senzora robota S3 spaja na pin 5 (ili pin PB2 mikrokontrolera)
 - na pinovima 2, 4 i 6 je spojeno +5V, odnosno na pinovima 8 i 10 je spojen GND, preko čega se napajaju svih pet infracrvenih senzora
- preko konektora „CON2“ se programira mikrokontroler putem SPI komunikacije, za što je potrebno šest pinova : MOSI, MISO, SCK, RESET, VCC i GND
- pinovi mikrokontrolera (PC0, PC2...PC7) su ostali neiskorišteni i spojeni su na muški konektor u obliku letvice
- na pinove (X6-1...X6-6) ženskog konektora u obliku letvice se spaja bluetooth modul HC-05, koji putem serijske komunikacije (UART) ostvaruje vezu sa μ C
 - X6-2 i X6-3 su pinovi preko kojih se ostvaruje serijska komunikacija, a pinovi X6-4 i X6-5 služe za napajanje bluetooth modula (+5 V)
 - pinovi X6-2 i X6-3 se spajaju na odgovarajuće pinove mikrokontrolera, a to su pinovi za UART serijsku komunikaciju PD0 (RXD) i PD1 (TXD)
 - otpornici R9 i R10 služe kao djelilo napona kako bi se dobilo 3.3 V na prijemnom pinu (Rx) bluetooth modula
 - ostali pinovi ženskog konektora (X6-7...X6-14) su ostali neiskorišteni
- H-most L293d se spaja s mikrokontrolerom prema slici 4.1., gdje:
 - pinovi mikrokontrolera PD4 (OC1B) i PD5 (OC1A) su PWM izlazi
 - pinovi mikrokontrolera PC1, PD2 i PD3, PD6 su digitalni izlazi kojim se mijenja smjer vrtnje lijevog, odnosno desnog motora

- na konektore X4-1, X4-2 i X3-1, X3-2 se spaja lijevi (Motor 2), odnosno desni motor (Motor 1) robota
- preko konektora X5-1 i X5-2 se napajaju oba DC motora, a ulazni napon mora biti u rasponu 4,5 V – 7 V
- na pinu 9 (RESET) mikrokontrolera je spojen „pull-up“ otpornik R1 (10 k Ω) i tipkalo, čijim pritiskom resetiramo program koji se izvršava u μ C
- na pinovima mikrokontrolera 12 (XTAL2) i 13 (XTAL1) je spojen kristalni oscilator koji daje takt procesoru ATmega32 mikrokontrolera, frekvencije 16 MHz
- pinovi mikrokontrolera (10, 11, 30...32) služe za napajanje mikrokontrolera (+ 5 V), a keramički kondenzator C3 (100 nF) služi za stabilizaciju napona

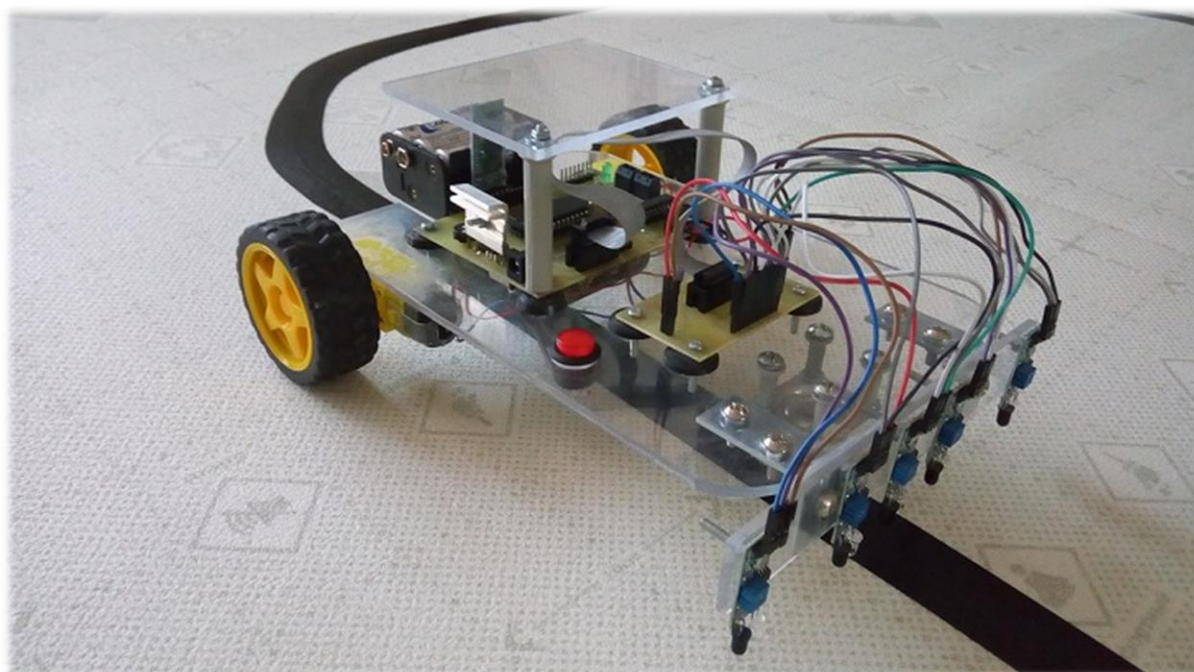
Fizički izgled tiskane pločice izrađene prema električnoj shemi na slici 4.1., prikazana je slikom 4.2.



Slika 4.2. Upravljačka jedinica mobilnog robota

5. MOGUĆA POBOLJŠANJA FUNKCIONALNOG I DIZAJNERSKOG SEGMENTA MOBILNOG ROBOTA

U ovome poglavlju žele se istaknuti nedostaci modela mobilnog robota u smislu njegove funkcionalnosti i dizajna. Pri gradnji modela mobilnog robota za slijeđenje linije prikazanog u ovom završnom radu, veća pažnja je dana funkcionalnom dijelu. Za početak se može pogledati tiskana pločica prikazana na slici 4.2. Električni vodovi tiskane pločice su predimenzionirani u odnosu na iznos struje koja prolazi vodovima. Za struju iznosa 330 mA preporuča se širina voda oko 90 μm (debljina voda 35 μm). Smanjivanjem dimenzija vodova, smanjili bi i dimenzije tiskane pločice koja bi tada bila praktičnija za ugradnju. *On/off* prekidač i tipkalo za resetiranje mikrokontrolera su loše pozicionirani na pločici, zbog čega je otežan pristup tim komponentama i nisu naklonjene korisniku. Iz toga razloga je *on/off* prekidač sa slike 4.2. zamijenjen velikim prekidačem koji je smješten na tijelo robota. Redne stezaljke na koje se priključuju motori bi se trebale zamijeniti čvršćim rednim stezaljkama ili drugim konektorima veće kvalitete. Ispod L293d *driver-a* bi trebao biti sloj bakra koji će hladiti element pri višim sklopnim frekvencijama. Tiskana pločica sa konektorima koji spajaju infracrvene senzore sa upravljačkom jedinicom bi trebala biti uklonjena, odnosno trebala bi se ostvariti jednostavnija veza. Za precizniju regulaciju moglo bi se u liniju dodati još IR senzora, ili koristiti senzor s mogućnošću mjerenja pozicije u finijoj podjeli. Nedostatak motora je malen raspon napona napajanja (3 V-6 V), odnosno manje su mogućnosti upravljanja brzinom vozila. Baterije koje napajaju motore i elektroniku vozila bi trebale biti zatvorene u određeno kućište, zaštićene od vanjskih utjecaja, a i zbog dizajna. U smislu dizajna svi električni vodovi i elektronički elementi bi trebali biti skriveni kako bi se mobilnom robotu dao kompaktan izgled. Izgled modela mobilnog robota dan je na slici 5.1.



Slika 5.1. Mobilni robot za slijeđenje linije

6. OPIS PROGRAMA (*Firmware*)

Programska podrška koja omogućava upravljanje, nadzor i manipulaciju podacima, a ugrađena je u elektronski uređaj (npr. mikrokontroler), naziva se *firmware*. Takav program se u mikrokontroleru pohranjuje u stalnu *flash* ili EEPROM memoriju. Program je pisan u staroj verziji integriranog razvojnog sučelja AVRStudio 4.18, koji je baziran na programskom jeziku C. U nastavku će se dati prikaz cijeloga programa, a zatim i njegov opis.

```
1  #include <stdio.h>
2  #include <avr/io.h>
3  #include <util/delay.h>
4  #include <math.h>
5  #include "UART.h"
6
7  #define F_CPU                16000000UL
8
9  #define sensor1              bit_is_clear(PINB,0)
10 #define sensor2              bit_is_clear(PINB,1)
11 #define sensor3              bit_is_clear(PINB,2)
12 #define sensor4              bit_is_clear(PINB,3)
13 #define sensor5              bit_is_clear(PINB,4)
14
15 unsigned int pGain = 1000 ;
16 float iGain ;
17 unsigned int dGain ;
18
19 int eInteg ;
20 int ePrev ;
21 volatile unsigned char z ;
22 volatile unsigned int n_Ref = 3199 ;
23
24 int  ReadSensors(void) ;
25 float PID(unsigned int req_value, int cur_value) ;
26 void  PWM_init(void) ;
27
28
29 int ReadSensors()
30 {
31     int sensor_Value ;
32
33
34     if (!(sensor1) && (sensor2) && (sensor3) && (sensor4) && (sensor5))
35         sensor_Value = 4 ;
36
37
38
39     if (!(sensor1) && !(sensor2) && (sensor3) && (sensor4) && (sensor5))
40         sensor_Value = 3 ;
41
42
43
44     if ((sensor1) && !(sensor2) && (sensor3) && (sensor4) && (sensor5))
45         sensor_Value = 2 ;
46
47
48
49     if ((sensor1) && !(sensor2) && !(sensor3) && (sensor4) && (sensor5))
```

```
50     sensor_Value = 1 ;
51
52
53
54     if ((sensor1) && (sensor2) && !(sensor3) && (sensor4) && (sensor5))
55         sensor_Value = 0 ;
56
57
58
59     if ((sensor1) && (sensor2) && !(sensor3) && !(sensor4) && (sensor5))
60         sensor_Value = -1 ;
61
62
63
64     if ((sensor1) && (sensor2) && (sensor3) && !(sensor4) && (sensor5))
65         sensor_Value = -2 ;
66
67
68
69     if ((sensor1) && (sensor2) && (sensor3) && !(sensor4) && !(sensor5))
70         sensor_Value = -3 ;
71
72
73
74     if ((sensor1) && (sensor2) && (sensor3) && (sensor4) && !(sensor5))
75         sensor_Value = -4 ;
76
77
78
79     if ((sensor1) && (sensor2) && (sensor3) && (sensor4) && (sensor5))
80     {
81         return 0xFF ;
82     }
83
84     if (!(sensor1) && !(sensor2) && !(sensor3) && !(sensor4) && !(sensor5))
85     {
86         return 0xFE ;
87     }
88
89
90     return sensor_Value ;
91
92 }
93
94
95
96 float PID(unsigned int req_value, int cur_value)
97 {
98     int error ;
99     float pid_value ;
100
101     error = req_value - cur_value ;
102
103     pid_value = (float) (pGain * error) + (iGain * eInteg) +
104                 (dGain * (error - ePrev)) ;
105
106     eInteg += error ;
107     ePrev = error ;
108
109     return pid_value ;
110
111 }
```

```
112
113
114 void PWM_init()
115 {
116     DDRD |= (1<<DDD4) | (1<<DDD5) ;
117
118     TCCR1A |= (1<<WGM11) | (1<<COM1A1) | (1<<COM1B1) ;
119
120     TCCR1B |= (1<<WGM12) | (1<<WGM13) | (1<<CS10) ;
121
122     ICR1 = 3199 ;
123
124     OCR1A = 0 ;
125
126     OCR1B = 0 ;
127 }
128
129
130 int main(void)
131 {
132     DDRB |= (0<<DDB0) | (0<<DDB1) | (0<<DDB2) | (0<<DDB3) | (0<<DDB4) ;
133     PORTB |= (0<<PORTB0) | (0<<PORTB1) | (0<<PORTB2) | (0<<PORTB3) |
134             (0<<PORTB4) ;
135     DDRD |= (1<<DDD2) | (1<<DDD3) | (1<<DDD6) ;
136     DDRC |= (1<<DDC1) ;
137     PORTC &=~ (1<<PORTC1) ;
138     PORTD |= (1<<PORTD2) | (1<<PORTD3) | (0<<PORTD6) ;
139
140     PWM_init() ;
141     UART_Init(9600) ;
142     sei() ;
143
144     unsigned int    ref_Position = 0 ;
145     int             prev_Position = 0 ;
146     int             left_Motor_Speed ;
147     int             right_Motor_Speed ;
148     int             position ;
149     float           regulator_Value ;
150
151     char str1[5] ;
152     char str2[20] ;
153
154
155     uint8_t i ;
156
157     for (i=0 ; i < 200 ; i++)
158     {
159         _delay_ms(10) ;
160     }
161
162     while(1)
163     {
164
165         position = ReadSensors() ;
166
167         if (position == 0xFF)
168         {
169             while((sensor1 && (sensor2) && (sensor3) && (sensor4) &&
170                    (sensor5))
171             {
172                 if (prev_Position > 0)
173                 {
```

```
174             OCR1A = 3100 ;
175             OCR1B = 0 ;
176         }
177
178         else
179         {
180             OCR1A = 0 ;
181             OCR1B = 3100 ;
182         }
183     }
184
185     position = ReadSensors() ;
186 }
187
188 if (position == 0xFE)
189 {
190     while(!(sensor1) && !(sensor2) && !(sensor3) && !(sensor4) &&
191           !(sensor5))
192     {
193         OCR1A = 0 ;
194         OCR1B = 0 ;
195
196         if (z == 1)
197         {
198             OCR1A = 3000 ;
199             OCR1B = 3000 ;
200
201             _delay_ms(500) ;
202             _delay_ms(500) ;
203
204             z = 0 ;
205
206             break;
207         }
208     }
209
210     position = ReadSensors() ;
211 }
212
213
214     regulator_Value = PID(ref_Position,position) ;
215
216     right_Motor_Speed = (int) round(n_Ref + regulator_Value) ;
217     left_Motor_Speed  = (int) round(n_Ref - regulator_Value) ;
218
219
220     if (left_Motor_Speed > 3199)
221     {
222         left_Motor_Speed = 3199 ;
223     }
224
225     if (left_Motor_Speed < 0)
226     {
227         left_Motor_Speed = 0 ;
228     }
229
230     if (right_Motor_Speed > 3199)
231     {
232         right_Motor_Speed = 3199 ;
233     }
234
235     if (right_Motor_Speed < 0)
```

```
236         {
237             right_Motor_Speed = 0 ;
238         }
239
240
241         OCR1B = right_Motor_Speed ;
242         OCR1A = left_Motor_Speed ;
243
244         sprintf(str1,"%d",position) ;
245         sprintf(str2,"%d", regulator_Value) ;
246
247
248         UART_TransmitString(str1) ;
249         UART_Transmit(0x09) ;
250         UART_Transmit(0x09) ;
251
252         UART_TransmitString(str2) ;
253         UART_Transmit(0x0D) ;
254         UART_Transmit(0x0A) ;
255
256
257
258
259
260
261         for (i=0 ; i < 3 ; i++)
262         {
263             _delay_ms(10) ;
264         }
265
266         prev_Position = position ;
267
268     }
269 }
270 ISR(USART_RXC_vect)
271 {
272     unsigned char temp ;
273
274     temp = UDR ;
275
276     if (temp == '+')
277     {
278         n_Ref += 120 ;
279
280         if (n_Ref > 3199)
281             n_Ref = 3199;
282     }
283
284     if (temp == '-')
285     {
286         n_Ref -= 120 ;
287
288         if (n_Ref < 2000)
289             n_Ref = 2000;
290     }
291
292     if (temp == 'd')
293         z = 1 ;
294 }
```

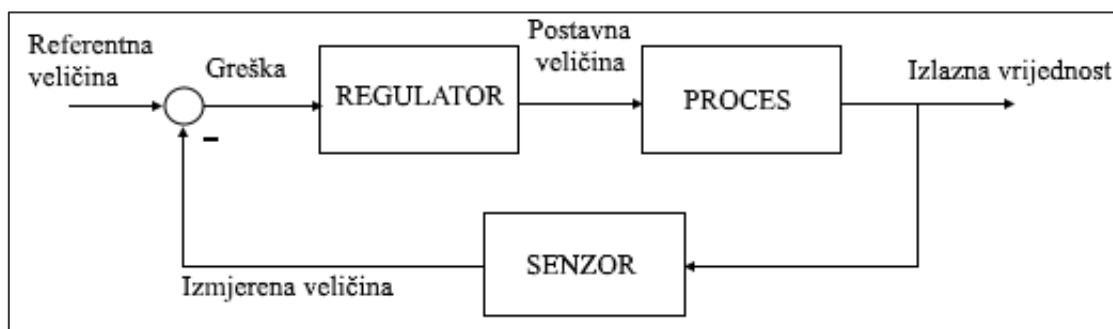

- na linijama 1-5 se uvrštavaju datoteke zaglavlja čije se funkcije koriste u programu (funkcije kašnjenja, serijska UART komunikacija, definiranje ulazno/izlazne jedinice mikrokontrolera, matematičke funkcije)
- linija 7 definira frekvenciju mikroprocesora 16 MHz
- na liniji 8-12 se nalaze makro definicije kojima se postiže lakše zapisivanje u daljnjem programu i preglednost koda
 - *sensorX* predstavlja senzor SX (slika 2.2.), za $X = \{1, 2, 3, 4, 5\}$
 - funkcije *bit_is_clear()* provjeravaju stanje ulaznih pinova, a u ovom slučaju to su prvih pet pinova port-a B (PINB0...PINB4)
 - npr. ukoliko je PINB0 u niskom stanju, *sensor1* postaje logička jedinica „1“
- na liniji 15-17 su deklarirane globalne varijable, a redom se odnose na proporcionalno, integralno i derivacijsko pojačanje (vrijednosti pojačanja određene su eksperimentalno, u ovom slučaju postavljen samo P regulator)
- na liniji 19 je deklarirana globalna varijabla u kojoj će se akumulirati greška (za integralno djelovanje), a na liniji 20 globalna varijabla koja pamti prethodnu grešku (za derivacijsko djelovanje)
- Na liniji 22 je deklarirana i inicijalizirana globalna varijabla koja se odnosi na referentnu brzinu motora. Ispred imena tipa varijable nalazi se ključna riječ *volatile* koja omogućuje promjenu vrijednosti varijable u potprogramu prekidne ISR rutine.
- na liniji 29 se nalazi funkcija *ReadSensors()* pomoću koje računamo poziciju robota
 - definirano je devet mogućih slučajeva pozicije robota, te je svakom slučaju dodijeljena određena težina (od -4 do +4)
 - na liniji 79 i 84 su definirani slučajevi kada robot ne vidi crnu liniju i kada su mu svi senzori na crnoj liniji
 - pozicija se sprema u lokalnu varijablu *sensor_Value*, a funkcija *ReadSensors()* se poziva u beskonačnoj petlji glavnog programa
- na liniji 96 je definirana funkcija *PID()* koja računa vrijednost izlaza regulatora na temelju izračunate greške
 - funkcija prima dva parametra (referentnu poziciju i izmjerenu poziciju), gdje njihova razlika čini grešku koja se sprema u lokalnu varijablu *error* (negativna povratna veza)
 - vrijednost izlaza regulatora se računa prema izrazu na liniji 103
- na liniji 114 se nalazi funkcija *PWM_init()* koja definira pwm izlaze mikrokontrolera na pinovima PIND4 i PIND5
 - na liniji 118 i 120 se definira brzi pwm, gdje se 16 bit-ni brojač resetira kada odbroji do vrijednosti zapisane u ICR1 registru
 - vrijednost zapisana u ICR1 registru definira frekvenciju pwm-a, a u ovom slučaju to je 5 kHz ($16000000 \text{ Hz} / 3200 = 5000 \text{ Hz}$)
 - vrijednost zapisana u 16 bit-nim registrima OCR1A i OCR1B definira faktor popunjenosti pwm-a, a moguće vrijednosti su u rasponu 0 – 3199
 - više o postavkama pwm izlaza može se vidjeti u [5] na str. 109
- na liniji 132 i 133 prvih pet pinova porta B se definiraju kao ulazni pinovi i onemogućuju im se ugrađeni *pull-up* otpornici

- izlazi senzora se spajaju na prethodno navedene ulazne pinove, gdje je srednji senzor robota S3 spojen na PINB2
- linije koda 135-138 definiraju izlazne pinove i njihova stanja kojima se upravlja L293d integriranim krugom, odnosno upravlja se smjer vrtnje oba motora
- na liniji 141 se poziva funkcija *UART_Init (9600)* koja omogućava serijsku UART komunikaciju pri brzini 9600 bps
 - funkcija se nalazi u samostalno napisanoj datoteci zaglavlja *UART.h* koja pretvara funkciju pinova D0 i D1 u Tx/Rx par
- na liniji 142 pomoću naredbe *sei()* se postavlja globalna interrupt zastava kako bi mogli omogućiti prekidnu rutinu UART komunikacije pri prijemu podataka koja se nalazi na liniji 270
- linija 144 definira lokalnu varijablu u kojoj se sprema vrijednost referentne pozicije i koristi se kao parametar *PID ()* funkcije
- Na liniji 146 i 147 se definiraju varijable u koje se spremaju vrijednosti koje predstavljaju brzinu lijevog (Motor 2), odnosno desnog motora (Motor 1). Te vrijednosti se ujedno upisuju u OCR1A i OCR1B registre.
- linija 148 definira lokalnu varijablu *position* u koju se sprema vrijednost, koju vraća funkcija *ReadSensors ()*, odnosno pozicija robota
- linija 149 definira lokalnu varijablu *regulator_Value* u koju se sprema vrijednost, koju vraća funkcija *PID ()*, odnosno vrijednost izlaza regulatora
- na linijama 151-152 se deklariraju dvije varijable tipa *char* u koje se redom upisuju pozicija robota i izlazna vrijednost regulatora
- *for* petlja na liniji 157 stvara kašnjenje od 2 sekunde pomoću naredbe *_delay_ms()*
- na liniji 167-186 definiran je slučaj kada mobilni robot skrene sa crne linije i senzori „vide“ samo bijelu podlogu
 - ovisno o prethodno izmjerenoj poziciji, jedan od motora će zakrenuti robot (drugi motor miruje) sve dok jedan od senzora ne primijeti crnu liniju
 - u registar OCR1A se zapisuje brzina lijevog motora, a u registar OCR1B brzina desnog motora
- na liniji 188-211 definiran je slučaj kada svi senzori „vide“ crnu podlogu
 - u istom trenutku mobilni robot stane (napon na motorima jednak nuli)
 - vozilo će krenuti samo kada se putem bluetooth komunikacije (npr. sa računala) pošalje slovo 'd'
 - bluetooth modul HC-05 koji se nalazi na robotu, komunicira sa mikrokontrolerom putem serijske UART komunikacije
 - vozilu se daje 1 sekunda da izađe sa crne podloge, te nakon toga senzori ponovno očitavaju poziciju robota
- kada se izmjerila pozicija robota, na liniji 214 se poziva funkcija *PID ()* koja vraća izlaznu vrijednost regulatora
- na liniji 216 i 217 izračunava se brzina lijevog (Motor 2), odnosno desnog motora (Motor 1) pomoću prethodno dobivene izlazne vrijednosti regulatora
- na liniji 220-238 se definira zasićenje napona za oba motora (0 - 3199), gdje vrijednost 3199 predstavlja maksimalan napon na motoru

- na liniji 244-245 pomoću funkcije *sprintf()* se varijable tipa *integer* pretvaraju u tip *string*, i pohranjuju u varijable *str1* i *str2*
- na liniji 261 definirana je *for* petlja koja uzrokuje kašnjenje od 30 ms
- na liniji 266 u varijablu *prev_Position* se sprema vrijednost trenutne pozicije, a u drugom prolazu programa (novom očitavanju senzora) vrijednost prethodne pozicije se koristi u slučaju kada mobilni robot potpuno izađe sa linije
- na liniji 248-258 se prethodno navedene varijable *str1* i *str2* šalju putem UART komunikacije na bluetooth modul
 - funkcija *UART_Transmit()* za slanje jednog alfanumeričkog znaka i funkcija *UART_TransmitString* za slanje *string*-a se nalaze u datoteci zaglavlja *UART.h*
 - po ASCII kodu, heksadecimalna vrijednost 0x09 predstavlja horizontalni tab, 0x0D vraća pokazivač na početak linije, a 0x0A predstavlja prelazak u novu liniju (\n)
- na liniji 270 je definiran prekidni potprogram koji će se izvršiti svaki puta kada prekidni vektor *USART_RXC_vect* postane aktivan, odnosno kada prijemni Rx pin mikrokontrolera dobije podatak od bluetooth modula
 - primljeni podatak se sprema u UDR (*USART data register*) registar, a na liniji 274 se podatak čita iz toga registra i sprema u varijablu *temp*
 - ako je '+' primljeni alfanumerički znak, brzina oba motora mobilnog robota će se povećati za približno 10 %
 - ako je '-' primljeni alfanumerički znak, brzina oba motora mobilnog robota će se smanjiti za približno 10 %
 - Alfanumerički znak 'd' se šalje samo kada se želi pokrenuti mobilni robot sa crne podloge, za što je kod definiran na liniji 188

7. ANALIZA REGULACIJE POZICIJE MOBILNOG ROBOTA

Automatska regulacija omogućava održavanje zadanog ili željenog stanja nekog procesa, protiv svih unutrašnjih i vanjskih poremećaja. Osnovni blok dijagram regulacijskog kruga prikazan je na slici 7.1. Zadatak automatske regulacije je postići izjednačenje izlazne veličine iz procesa sa ulaznom referentnom veličinom. Regulacija izlazne procesne veličine se ostvaruje pomoću negativne povratne veze. Izlazna veličina se mjeri senzorom, te se šalje povratnom vezom kako bi se usporedila sa ulaznom referentnom veličinom. Ukoliko postoji razlika između izlazne i ulazne veličine, dobiti će se greška na temelju koje odgovarajući regulator daje postavnu veličinu. Postavnom veličinom regulator mijenja stanje procesa, odnosno regulira izlaznu veličinu sve dok greška ne bude jednaka nuli. U konkretnom slučaju objekt regulacije (proces) sa slike 7.1. predstavlja matematički model mobilnog robota. Blok nazvan „senzor“ u ovome slučaju predstavlja pet infracrvenih senzora. Regulatori su najčešće implementirani u program (*firmware*), pa govorimo o digitalnom regulatoru. Takav program je prikazan u poglavlju 6. Izlazna veličina procesa u ovome je slučaju pozicija mobilnog robota, dok je fiksna referentna veličina pozicije postavljena u programu. U sljedećim potpoglavljima će se izložiti analiza regulacije pozicije mobilnog robota za slijeđenje linije, s obzirom na promjenu vrste regulatora i promjenu brzine mobilnog robota. Bitno je napomenuti da u ovom završnom radu nije izvršena sinteza regulatora prema nekoj od postojećih teoretskih metoda zbog nepoznavanja matematičkog modela mobilnog robota. Vrijednosti parametara regulatora su birani nasumično, sve dok se eksperimentalnom provjerom nije dobilo željeno ponašanje robota. Ovom analizom se samo želi istaknuti razlike između proporcionalnog, integralnog i derivacijskog djelovanja, kako oni pojedinačno i u kombinaciji utječu na stanje reguliranog procesa.



Slika 7.1. Elementi osnovnog regulacijskog kruga

7.1. P regulator

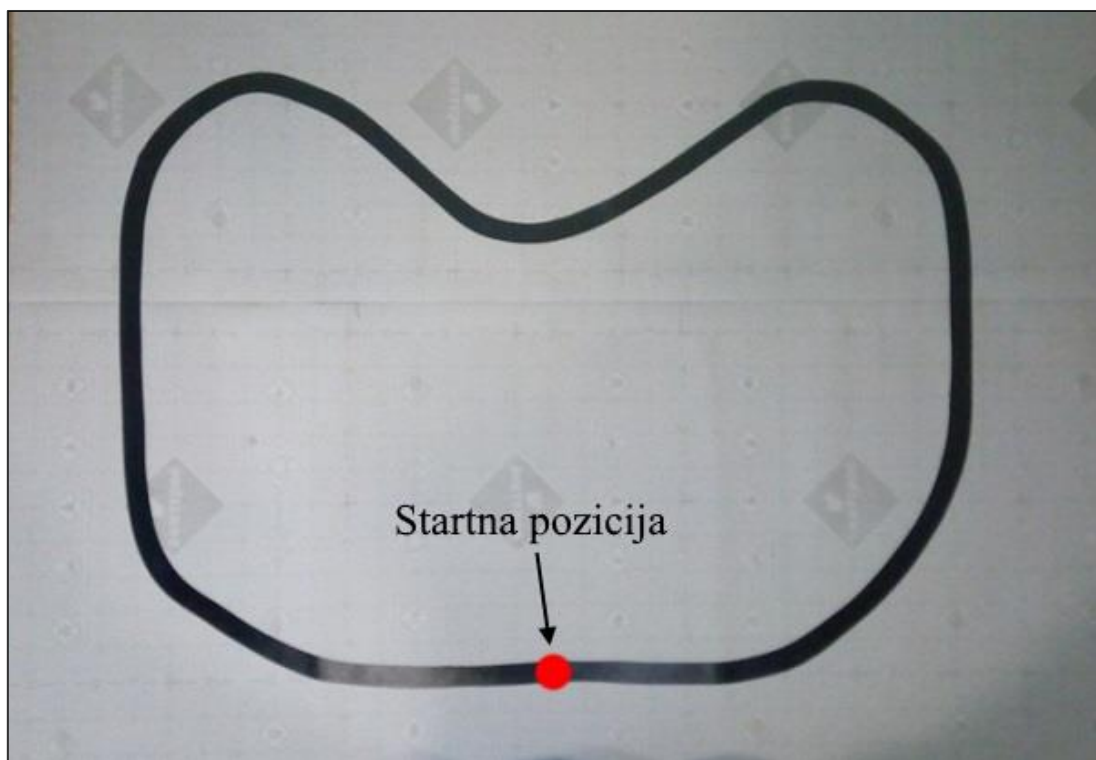
Proporcionalni regulator ima najjednostavnije djelovanje, koji daje izlazni signal proporcionalan iznosu greške (razlika referentne i mjerene veličine). Ovaj regulator ne unosi kašnjenje u sustav, već naprotiv djeluje trenutno na bilo koji iznos greške. Stoga, dobra strana P regulatora je da doprinosi brzini odziva regulirane veličine, odnosno smanjuje vrijeme porasta. Proporcionalni regulator ne može poboljšati druge bitne značajke automatske regulacije kao što su postotni prebačaj, vrijeme smirivnja i oscilacije u odzivu. Formula proporcionalnog regulatora u vremenskoj domeni glasi :

$$y(t) = K_p e(t) , \quad (7.1)$$

gdje je:

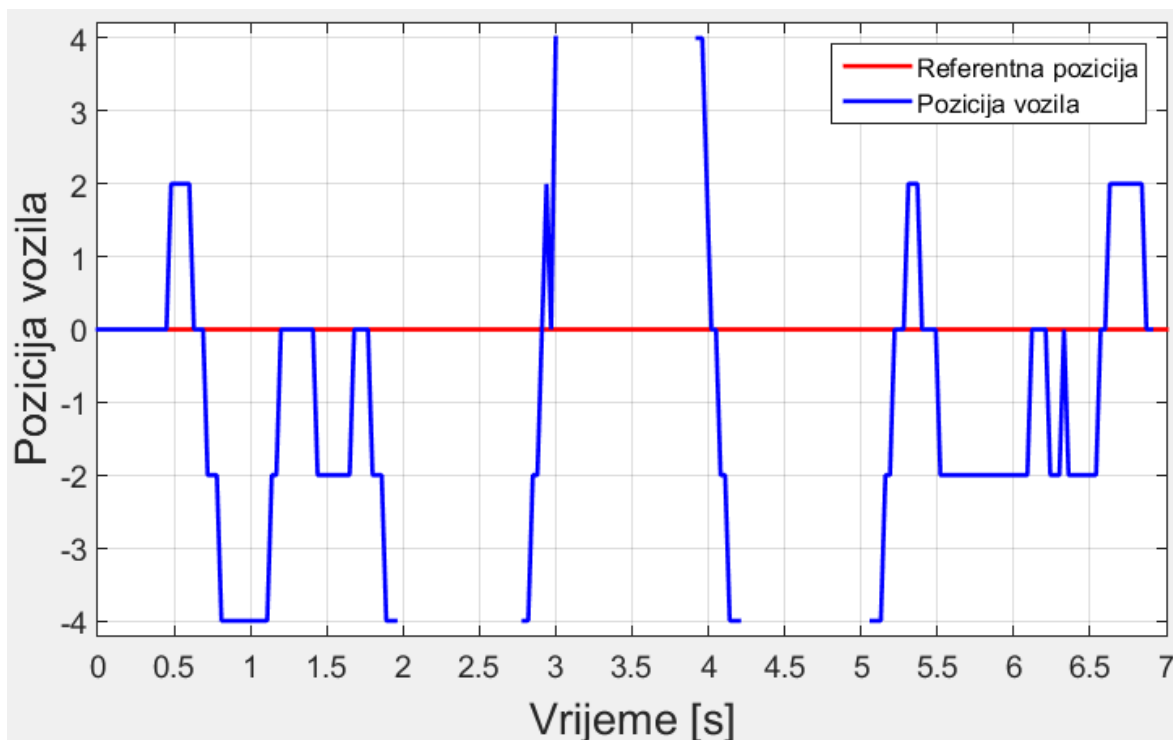
- $y(t)$ - izlazna vrijednost regulatora,
- K_p - proporcionalno pojačanje,
- $e(t)$ - greška (razlika referentne i izmjerene vrijednosti).

Na temelju dobivenog optimalnog ponašanja robota, određen je parametar $K_p = 1000$ s kojim je provedeno mjerenje. Na slici 7.3. prikazan je snimljeni odziv regulirane pozicije mobilnog robota pri njegovoj maksimalnoj brzini, a na slici 7.4. prikazane su izlazne vrijednosti P regulatora. Mjerenje se provodi na stazi koja je prikazana na slici 7.2., a izmjerene vrijednosti se odnose na jedan puni krug.

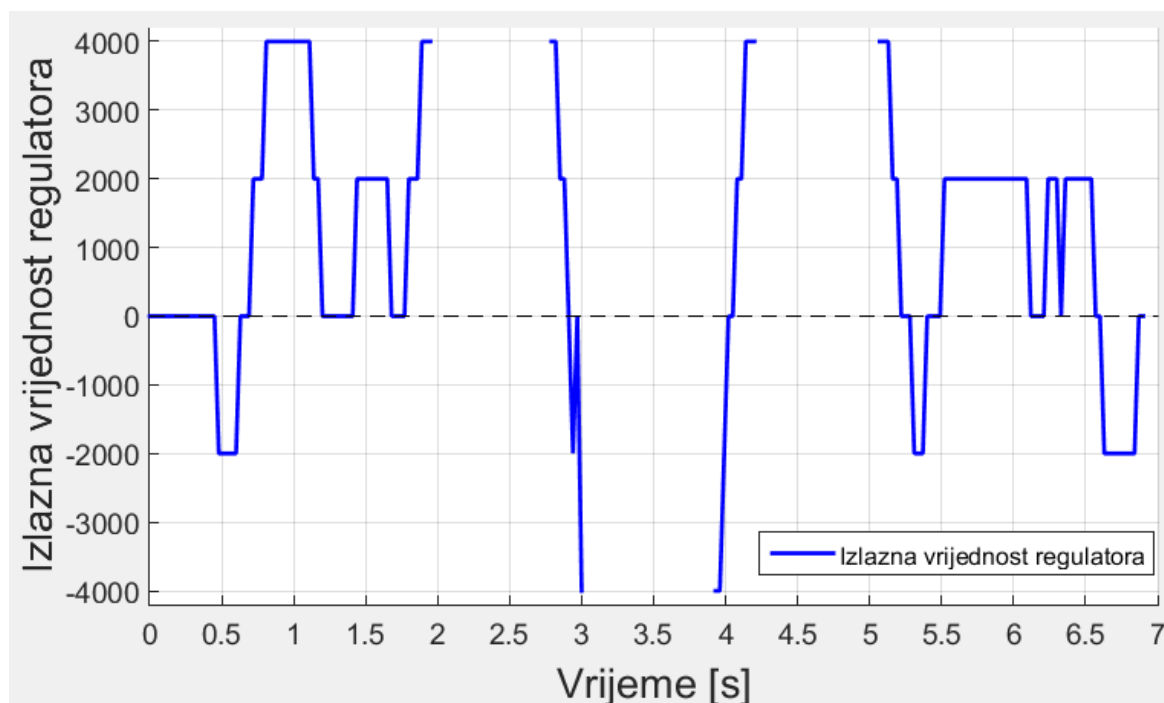


Slika 7.2. Staza na kojoj se provodi mjerenje

7.1.1. Regulacija pozicije pri maksimalnoj brzini vozila

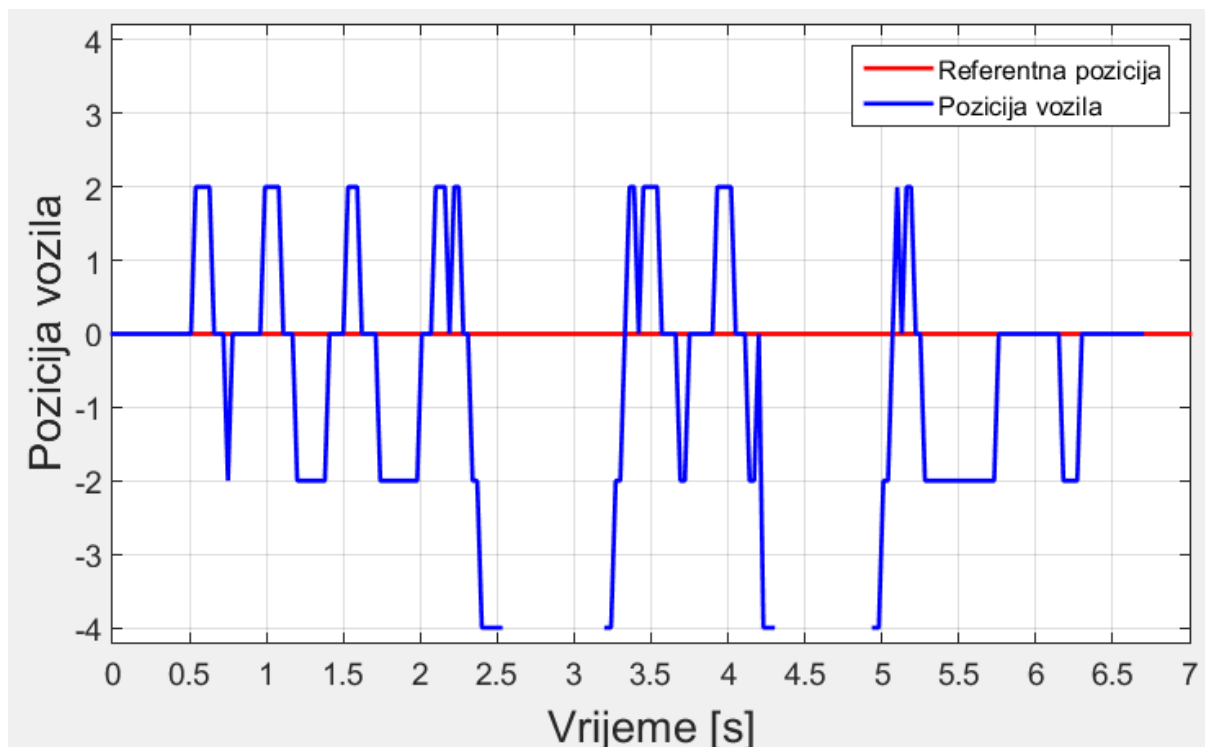


Slika 7.3. Pozicija vozila sa P regulatorom (maksimalna brzina vozila)

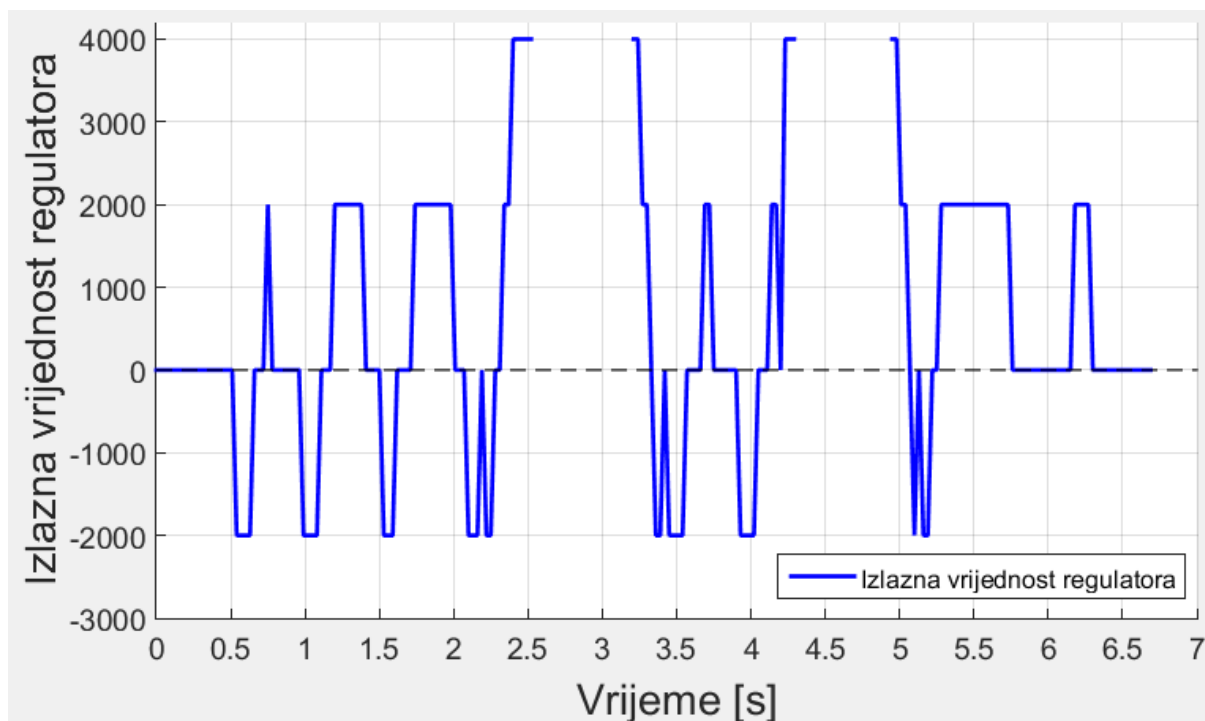


Slika 7.4. Izlazna vrijednost P regulatora (maksimalna brzina vozila)

7.1.2. Regulacija pozicije vozila (40 % manja brzina)



Slika 7.5. Pozicija vozila sa P regulatorom (40 % manja brzina vozila)



Slika 7.6. Izlazna vrijednost P regulatora (40 % manja brzina vozila)

Prilikom određivanja parametra proporcionalnog regulatora potrebno je odabrati rješenje koje će osigurati veliku brzinu odziva vozila, sa što manjim oscilacijama. Mali iznosi pojačanja K_p će dati tromo slijeđenje putanje sa malim oscilacijama, dok će veliki iznosi pojačanja osigurati brže reakcije vozila, ali i veće oscilacije. Na temelju dobivenih rezultata ne može se dati ozbiljna analiza djelovanja regulatora zbog nemogućnosti mjerenja pozicije u finijoj podjeli korištenim senzorima. Dana je usporedba regulacije pozicije pri većoj i manjoj brzini mobilnog robota. Cilj je zadržati vozilo u području pozicija od -2 do 2. Na prethodnim slikama se mogu vidjeti prekidi kod izmjerenih pozicija vozila. Mjesta prekida označavaju da je mobilni robot potpuno izletio sa zadane putanje, odnosno kada nijedan senzor ne registrira crnu liniju. U tim situacijama djelovanje regulatora se isključuje, daje se napon određenog iznosa na lijevi ili desni motor vozila (ovisno o situaciji) kako bi se robot zarotirao, i ponovno pronašao putanju. Može se zaključiti da je regulacija bolja pri manjoj brzini vozila, manji su prebačaji i manje su oscilacije vozila. Vidi se da je izlazna vrijednost P regulatora proporcionalna razlici referentne i stvarne pozicije robota. No i pri smanjenoj brzini vozila, proporcionalni regulator nije uspio potpuno zadržati vozilo na zadanoj putanji kod oštrijih promjena smjera. Kako bi se poboljšalo slijeđenje zadane putanje, u nastavku će se proporcionalnom djelovanju dodati derivacijsko djelovanje.

7.2. PD regulator

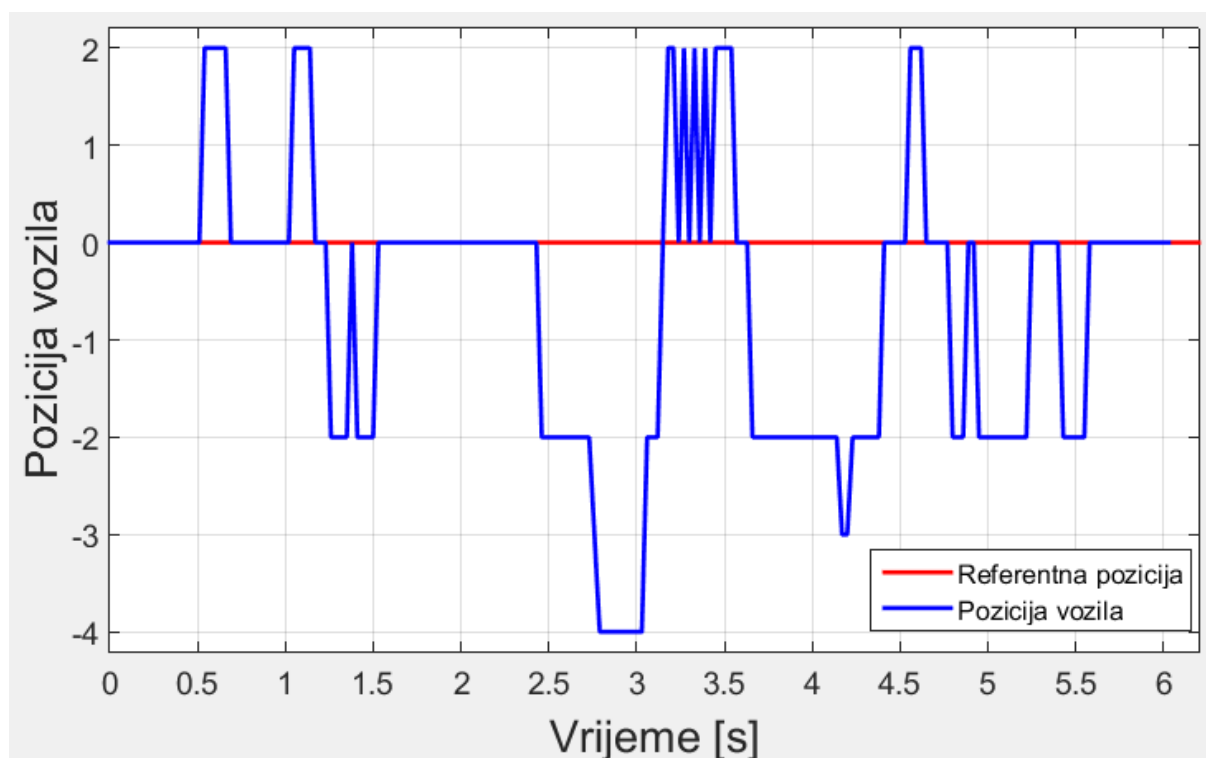
Iako je PI regulator najčešće korišten, za praćenje putanje i općenito za regulaciju gibajućih objekata bolje rezultate daje PD regulator. Ovdje je u interesu što brža reakcija vozila na promjene smjera putanje, što derivacijsko djelovanje u kombinaciji sa proporcionalnim i može osigurati. Kada se na formulu (6.1) doda derivacijski član, dobivamo matematički oblik PD regulatora u vremenskoj domeni:

$$y(t) = K_p e(t) + K_d \frac{de(t)}{dt}, \quad (7.2)$$

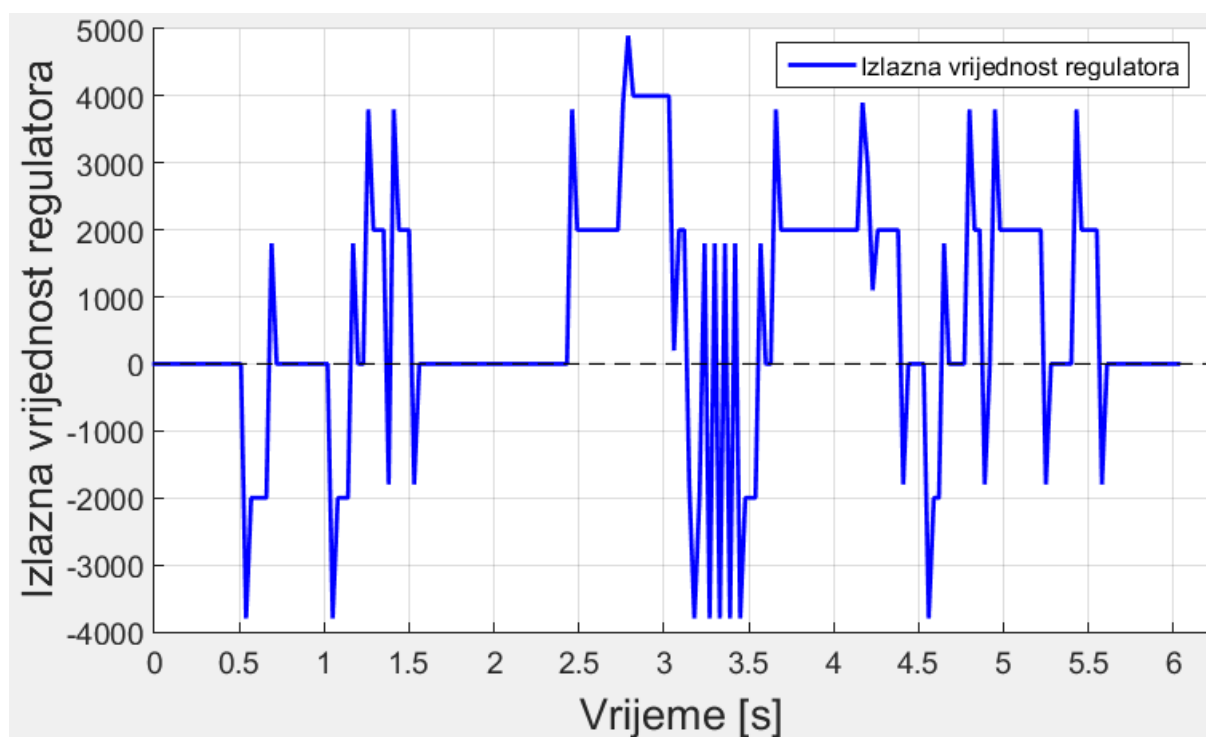
gdje je:

$$\begin{aligned} K_d & \text{ - derivacijsko pojačanje,} \\ \frac{de(t)}{dt} & \text{ - iznos promjene greške } e \text{ u periodu } dt. \end{aligned}$$

Iz formule (7.2) se vidi da je iznos derivacijskog djelovanja proporcionalan brzini promjene greške u intervalu dt , zbog čega derivacijski član ima prediktivno djelovanje. Za razliku od proporcionalnog, derivacijski član ne djeluje samo na iznos greške, nego i na iznos promjene greške. Derivacijski član nastoji djelovati protiv proporcionalnog, čime unosi efekt prigušenja, smanjuje prebačaj i oscilacije regulirane veličine. Iznos derivacijskog pojačanja K_d je potrebno odabrati tako da vozilo ima brze reakcije na promjene smjera putanje, ali i da prigušenje ne bude veliko, kako ne bi dobili trome reakcije vozila. Na temelju prethodna dva kriterija odabrano je pojačanje $K_d = 900$, dok je proporcionalno pojačanje ostavljeno na istom iznosu $K_p = 1000$. Na slici 7.7. je prikazan odziv regulirane pozicije vozila, a na slici 7.8. izlazne vrijednosti PD regulatora.



Slika 7.7. Pozicija vozila sa PD regulatorom



Slika 7.8. Izlazna vrijednost PD regulatora

Prethodna mjerenja su provedena pri brzini vozila od 60 % maksimalne brzine. Ako se usporede izmjerene pozicije na slikama 7.5. i 7.7., može se vidjeti da je PD regulator dao bolje ponašanje robota. Nema više potpunih izleta vozila sa staze i manje su oscilacije. Ako se pogledaju izlazne vrijednosti PD regulatora, mogu se uočiti njegova kratkotrajna forsiranja. Upravo ti „šiljci“ dolaze od derivacijskog djelovanja, kojima se mobilni robot prvotno želi što prije vratiti na stazu, a „šiljak“ koji se nalazi s druge strane x osi osigurava prigušenje, odnosno smanjuje prebačaj vozila preko referentne pozicije. U vremenu od 3 - 3.5 s, na dijelu staze koji je valovit se dogodio vrlo dinamičan izlaz PD regulatora. To se dogodilo zbog niske rezolucije mjerenja pozicije korištenim senzorima, zbog čega se mjerena pozicija u tom trenutku dinamično mijenjala. Bitno je naglasiti da PD regulator ne može osigurati precizno pozicioniranje i ukloniti stacionarnu pogrešku, jer pri malim promjenama greške i iznos derivacijskog djelovanja je neznatan.

7.3. PID regulator

Ako se PD regulatoru doda integralno djelovanje, može se ukloniti stacionarna greška i osigurati precizno pozicioniranje mobilnog robota na zadanu putanju. I - član akumulira grešku sve dok je ona prisutna. Integralni član najučinkovitije djeluje na male iznose greške kada je regulirana veličina blizu referentne, dok za velike iznose greške ima sporo djelovanje. Brzina odziva, prebačaj i oscilacije se ne mogu poboljšati integralnim djelovanjem, već se samo može eliminirati stacionarna greška. Formula PID regulatora u vremenskoj domeni glasi:

$$y(t) = K_p e(t) + K_d \frac{de(t)}{dt} + K_i \int_0^t e(t) dt, \quad (7.3)$$

gdje je:

K_i - integralno pojačanje.

U ovoj analizi provedeno je mjerenje i sa PID regulatorom, ali se nije vidjelo poboljšanje u slijeđenju zadane putanje. Glavni razlog je gruba diskretizacija pozicije mobilnog robota, zbog čega integralno djelovanje ne može doprinijeti. Korištenjem senzora veće rezolucije, učinak integralnog djelovanja bi se sigurno vidio.

8. ZAKLJUČAK

Pokazalo se da je za regulaciju pozicije mobilnog robota zadovoljavajuće rješenje dao PD regulator. Prednost proporcionalno-derivacijskog regulatora je što omogućuje brze reakcije mobilnog robota pri ostrim promjenama smjera putanje bez velikih oscilacija, za razliku od PI regulatora koji može dati dobro ponašanje vozila samo na ravnim stazama i blagim zavojima. Stoga se vrsta regulatora može birati i prema obliku staze koju robot mora slijediti. Ukoliko nisu postavljeni visoki zahtjevi za precizno pozicioniranje robota prilikom slijeđenja linije, PD regulator može osigurati dobro ponašanje mobilnog robota. Ako se uz brze reakcije, mali prebačaj i male oscilacije želi osigurati i eliminacija stacionarne pogreške, koristi se PID regulator. Osim programa (*firmware*), veliku ulogu na kvalitetu regulacije imaju odabrani senzori. Senzori daju informaciju kako se mijenja regulirana veličina, stoga je od velike važnosti odabrati odgovarajuće senzore. Infracrveni senzori modela mobilnog robota prikazani na slici 5.1., mogu samo dati vrlo diskretne vrijednosti pozicije, zbog čega je testiranje utjecaja integralnog člana na ponašanje vozila bilo onemogućeno.

LITERATURA

- [1] Šurina, T.: Automatska regulacija, Školska knjiga, Zagreb 1987.
- [2] Zorc, D.: Mikroprocesorsko upravljanje, predavanja, Fakultet strojarstva i brodogradnje, 2015.
- [3] Positive voltage regulator ICs, www.st.com/resource/en/datasheet/178.pdf, pristup 6.6.2017.
- [4] L293x Quadruple Half-H Drivers, <http://www.ti.com/lit/ds/symlink/l293.pdf>, pristup 7.6.2017.
- [5] Atmel AVR ATmega32 microcontroller, <http://www.atmel.com/images/doc2503.pdf>, pristup 9.6.2017.

PRILOZI

I. CD-R disc